PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 7/38, H01L 25/00, H03K 19/177	A1	(11) International Publication Number: WO 98/55918 (43) International Publication Date: 10 December 1998 (10.12.98)

(21) International Application Number: PCT/US98/11440

(22) International Filing Date: 3 June 1998 (03.06.98)

(30) Priority Data:

08/869,201	4 June 1997 (04.06.97)	US
08/978,691	26 November 1997 (26.11.97)	US

(71) Applicant: DYNACHIP CORPORATION [US/US]; 1255 Oakmead Parkway, Sunnyvale, CA 94086 (US).

(74) Agent: FISH, Ronald, Craig; Falk & Fish LLP, 16590 Oak View Circle, Morgan Hill, CA 95037 (US).

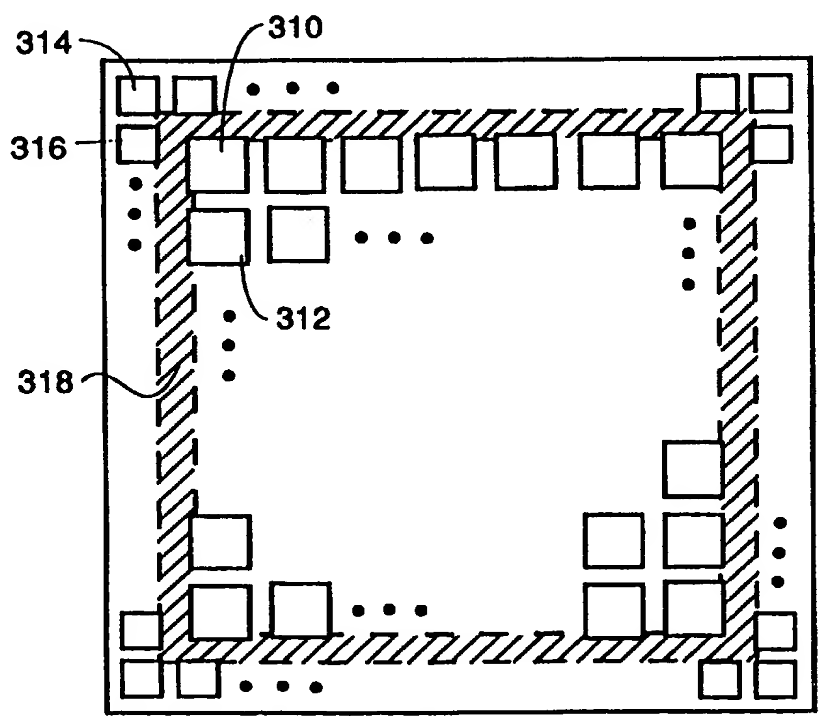
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: FPGA WITH CONDUCTORS SEGMENTED BY ACTIVE REPEATERS

(57) Abstract

An interface circuit for use in the layout of padframe interface circuits (318) for field programmable gate arrays having a plurality of I/O cells (314, 316), each of which may be programmed as an input or an output (or both) and a programmable connection matrix which provide programmable pathways between the data output signals generated by the core array of logic blocks (310, 312) and I/O cells (314, 316) programmed as outputs and provide programmable pathways between I/O cells (314, 316) programmed as inputs and data input conductors going into the core array (310, 312). Each interface circuit (318) also includes at least one and preferably two open spaces into which conductive paths may be laid out to carry power to the core array (310, 312) or carry dedicated signals to circuits other than the core which also reside on the integrated circuit.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

FPGA WITH CONDUCTORS SEGMENTED BY ACTIVE REPEATERS

BACKGROUND OF THE INVENTION

Logic designers have long had the need for custom logic circuits to implement their designs. In the 1970's, this need gave rise to programmable logic arrays, programmable array logic and programmable read only memory. Later in the decade, custom circuits were made by customizing the metal layer of integrated circuits which had standard cells formed in the layers below the metal layer. The customized metal layer interconnected the standard cells in a manner defined by the customer of the gate array manufacturer.

Gate arrays are only a good choice where the desired function to be performed by the gate array can be determined with certainty in advance. However, gate arrays are not a good choice where the desired function can change over time with changing requirements. This can happen when a circuit design is being evaluated and testing over time reveals the need for changes in the design. Another shortcoming of gate arrays was that they could not perform the function of packet encapsulation and delivery in network settings where packet construction was subject to a variety of different protocols and where packet headers change as the packets circulate, for example in token ring networks. This function has been done in software in the prior art, but increasing network speeds demands more speed which requires that this function be done in hardware.

The market for field reprogrammable gate arrays (FPGAs) has been increasing steadily in recent years, because of the advantages they provide to designers. FPGAs comprise an array of standard cells which include certain high usage types of logic items such as multiplexers, combinatorial logic etc. The standard cells can be programmed to implement whatever function the designer needs by setting certain programming bits. There are different technologies used to implement these programming bits. One technology is called antifuse where small fuse-like structures in various lines are selectively burned out to "cut" connection lines to get the desired functionality. Other technologies include static RAM (SRAM) based technologies like the "pass transistor" interconnect type FPGAs marketed by competitors of the assignee and the "active repeater" type FPGAs described herein. In SRAM based FPGAs, thousands of programming bit are stored in SRAM cells spread throughout the FPGA. These programming bits control the conductivity state of the channel of pass transistors or the on-off state of active repeaters inside each standard cell and in connection matrices connecting the standard cells to each other and to the package pins. By properly

programming all the SRAM cells, it is possible to make the necessary connections to implement the desired logic function.

The advantage of these FPGA structures is that a designer can have a custom design implemented in silicon much more rapidly than would be the case for a full custom chip design from scratch. Further, since the programming is not permanent (except in antifuse FPGAs), the functions implemented in the chip can be rapidly changed as the designer's design changes.

Routability is a key factor in FPGA commercial success. This means that for complex FPGAs, with many input and output signals and many needed connections between the logic blocks of the core array, there is a need for the software that programs the programmable connections to find individual routes for each needed interconnection without conflicts with other signals. Two different signals cannot share the same connection path absent some form of multiplexing. The more complex the functionality desired is, the more difficult the routing problem becomes and the more complex is the routing software.

Routability is not a significant problem in antifuse technologies since the antifuse programming connections are so small in terms of die area consumed per programmable connection that connection matrices can be fully populated (one programmable switch at every intersection of crossing wires) and many more options are available for making connections. However, antifuse has many problems which have caused designers to move away from antifuse FPGAs. First, antifuse structure require special processes to construct. These special processes complicate the manufacturing process. Another problem is that the programming voltage necessary to burn out an antifuse structure to program it is well above the Vcc voltage needed to run the rest of the circuit. This requires either a separate power supply, which customers do not like, or circuitry on the FPGA to generate the high programming voltage (typically 10 volts) from the low Vcc voltage (typically 3 volts). This circuitry complicates the design an uses chip area. Antifuse structures also take a long time to program each one, and since there can be hundreds of thousands of them on the chip, the total programming time can extend for a long time, which is bad for high volume production. Further, antifuse structures, once programmed, cannot be reprogrammed. If the design changes, an entirely new FPGA must be programmed for the new design. Finally, antifuse structures do not scale well as new technologies with smaller linewidths surface.

As a result of these problems with antifuse structures, the industry has tended to gravitate toward the SRAM based FPGAs noted above. SRAM based FPGAs however have their own set of problems which revolve around routability. SRAM cells are much

larger than antifuse structures, and since there are many thousands or hundreds of thousands of them on an FPGA, considerable die area can be consumed just by the memory cells needed to program the necessary interconnects. As a result, connection matrices as the intersections of two or more multiconductor interconnect buses cannot be fully populated with a switch and its associated SRAM cell at the intersection of each individual conductor with each other individual conductor. This means that fewer connection possibilities are available to the routing software. As a result, no SRAM based FPGA company can guarantee routing for every desired functionality specified by a company.

The routing problem of an SRAM based FPGA breaks down into two problems: routability within the core array to make all the necessary connections between different logic blocks within the core array; and routability of connections through the padframe interface area from the core array and the I/O cells. Of these two problems, routing within the core is the more difficult of the two. Routing bottlenecks can occur where the number of needed connections to a logic block or to an I/O cell exceeds the number of possible programmable pathways to the I/O cell.

SUMMARY OF THE INVENTION

In a second invention, each of the individual conductors of the vertical and horizontal buses that pass through the core logic array are segmented into multiple segments by active repeaters. The terms "active repeater" and "active link" mean the same thing and include both unidirectional and bidirectional active devices that can bring gain to bear and which provide no "direct connection" between line segments coupled to the active repeater (as the term "direct connection" is defined later herein-basically it means no direct conductive path for holes or electrons to flow from one line segment to the other through the active repeater). This lack of a "direct connection" provides isolation between the line segments such that the parasitic capacitances coupled to one line segment are not charged by drivers coupled to the other line segment but are charged by the active repeater which separates the two segments. This substantially speeds up performance by reducing delays and rendering the delays more predictable when the routing software does its routing. Because these active repeaters do not provide direct paths by which charge carriers in one line segment coupled to the input of an active repeater can reach parasitic capacitances coupled to line segments coupled to the output of the active repeater and because the active repeater has gain, several advantages flow. First, the lack of a direct connection causes the total parasitic capacitance load of each line segment to be fixed regardless of the programming state of the active repeaters. This means that the drivers coupled to that line segment can be optimized in structure to handle that exact capacitive load since it will not change

regardless of how many other line segments are coupled to it to make an interconnect. Second, segmentation means different line segments of the same line can be used to make different interconnects by the routing software. Further, the gain of the active repeaters means that degradation of the signals propagating through a chain of active repeaters can be minimized or eliminated since losses to parasitics can be replaced by energy from the power supply driving the active repeater. A further advantage is achieved by staggering the active repeaters in the vertical and horizontal buses. Staggering the repeaters such as in echelon fashion increases "access", i.e., the number of logic blocks a logic block output can be connected to by passing through only one vertical-to-horizontal active repeater and no repeater boundaries between two line segments on the same line. Higher degrees of access further simplify the routing software. The same techniques of using active repeaters and staggering them are used in the RIUs in the I_lines and O_lines.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is diagram illustrating a typical prior art FPGA with passive link structure in the form of an MOS pass transistors.

Figure 2 is an equivalent circuit showing the parasitic components of the prior art structure of Figure 1.

Figure 3 is a cross section of a typical prior art antifuse structure.

Figure 4 is an equivalent circuit showing the parasitic elements of the prior art antifuse structure of Figure 3.

Figure 5 is diagram illustrating a symbolic structure for a field programmable gate array using the active link technology disclosed herein.

Figure 6A is a circuit diagram of a typical NMOS inverter active link using a saturated MOS transistor for a load resistance.

Figure 6B is a circuit diagram of a typical NMOS inverter active link using a passive resistor for a load resistance.

Figure 6C is a circuit diagram of a typical NMOS inverter active link using a nonsaturated MOS transistor operating in the triode region for a load resistance.

Figure 6D is a circuit diagram of a typical NMOS inverter active link using a depletion mode MOS transistor for a load resistance.

Figure 7 is a circuit diagram for an active link in the form of an NMOS inverter having push-pull outputs and a tristate circuitry to improve the isolation between the lines to be connected.

Figure 8 is a circuit diagram for an active link in the form of a NAND gate.

Figure 9 is a truth table showing how the NAND gate of Figure 7 is used as an active link.

Figure 10 is a circuit diagram of a CMOS implementation of a NAND gate.

Figure 11 is a bidirectional, high input impedance active link design using emitter followers to provide high input impedance.

Figure 12 is a symbolic diagram of an FPGA using both long range and short range active links.

Figure 13 is a circuit diagram of an active link in the form of a biCMOS, differential, current-mode, one-input-two-output multiplexer.

Figure 14 is a circuit diagram for an active link biCMOS differential, current-mode, two-input-single-output multiplexer.

Figure 15 is a circuit diagram of another notation for the multiplexer of Figure 14 having four inputs and a single output.

Figure 16 is a circuit diagram for a crossbar switch having four inputs and three outputs using the notation of Figure 15.

Figure 17 is a circuit diagram of a typical set of high speed emitter followers for use with the circuits of Figures 13-16 as a driver to provide high input impedance or as a stand-alone, field-programmable active link.

Figure 18 is a circuit diagram of one example of how the structures of Figures 13-16 can be implemented in single ended ECL technology.

Figure 19 is an overall top view of the layout of an FPGA showing the relative position of the RIUs according to the teachings of the invention.

Figure 20 is a block diagram of a typical logic block such as logic block 312 that makes up the core array.

Figure 21 is a more detailed diagram of the programmable connection paths within the core array of logic blocks and showing the columns of input and output lines such as column 350 that exit the core array for connection to the RIUs.

Figure 22A is a schematic diagram of the preferred tri-state driver for use in the programmable connection matrices, and Figure 22B is a truth table showing the operation of the tri-state driver of Figure 22A.

Figure 23A is a circuit diagram of the preferred CMOS bidirectional tri-state driver, and Figure 23B is a truth table defining the operation of the circuit of Figure 23A.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Prior art FPGAs using pass transistor interconnect technology have been slow. The reason for this is because of the interconnect technology. Using standard 0.35

micron line widths, the logic blocks can run at speeds in excess of 200 MHz. The I/O structure speed depends upon the package speed, loading on the pins and output voltage swings, but with a high speed package and good board design, the I/O structure can run at speeds of 100 MHz. This leaves the interconnect structure as the key culprit and bottleneck factor in causing low speed performance of FPGAs.

If all the interconnects that need to be made in a typical FPGA could be made point-to-point, the FPGA could run at 250 MHz. The software that routes the interconnects however cannot make all interconnections point-to-point because of the not fully populated connection matrices of SRAM based FPGAs. As a result, a few of the interconnects are quite long and pass through many programmable pass transistors to make the appropriate switching connections in prior art FPGAs. Each pass transistor in these prior art structures adversely affects the signal propagating on the interconnect by subjecting it to parasitic RC networks to be described below. These parasitic RC networks slow propagation of the signals on these long interconnect "slow nets" enough and act as a bottleneck on the overall speed of the FPGA. Further, the V_t drop of each pass transistor means that there is less and less voltage after each pass transistor to charge up parasitic capacitances which further slows down propagation over the degradation due to the RC time constants alone. Speeding up the fast nets (point-to-point connections) does not increase performance, but speeding up these slow nets will increase performance. The active repeaters disclosed herein will speed up the slow nets.

Referring to Figure 1, there is shown a diagram of a typical field programmable gate array of the type using prior art passive links in the form of pass transistors such as is typically found in Xilinx FPGAs. To implement a desired function in this prior art FPGA, the output of a first standard cell 10 needs to be coupled to the input of a second standard cell 12. The FPGA integrated circuit has a matrix 14 of intersecting conductors on different layers which are insulated from each other. The conductors on one layer are symbolized by the vertical lines 16, 18, 20 and 22, while the conductors on another layer are symbolized by horizontal lines 24 and 26. To make the desired connection, 5 passive MOS "pass" transistors 28, 30, 32, 34 and 36 are programmed to make connections between various of the conductors in matrix 14. Pass transistor 28 is programmed to make a connection between the output 38 of standard cell 10 and matrix conductor 16. This is done by programming a logic 1 (or logic 0 depending upon the type of MOS transistor technology used for the pass transistors and logic levels) into memory cell 40.

Likewise, pass transistor 30 is programmed to make a connection between lines 16 and 24 by programming a logic 1 into memory cell 42. Pass transistor 32 is programmed to make a connection between lines 24 and 18 by programming a logic 1 into memory cell 44, and pass transistor 34 is programmed to make a connection between lines 18 and 26 by programming a logic 1 into memory cell 46. Finally, pass transistor 36 is programmed to make a connection between line 26 and the input 22 to standard cell 12 by programming a logic 1 into memory cell 48.

Because pass transistor 50 is not needed to make the connection between standard cell 10 and standard cell 12, it's memory cell 52 is programmed with a logic 0 so that no connection is made between line 16 and line 26 (unless this connection is needed as part of another conductive path between two different standard cells.

Memory cell 40, like all the other memory cells symbolized by an M in a box on the drawing, is typically a 6 transistor CMOS static RAM cell, and usually is part of a shift register (not shown) which snakes through the integrated circuit layout. This shift register contains the programming bits needed to control each pass transistor as well as any enable transistors in the standard cells. In alternative embodiments, the memory cells such as memory cell 40 can be stand alone memory cells coupled by shared data, address and control lines that go to each cell or memory cells grouped into a memory array having shared data input, address and control lines and coupled to the individual pass transistors by individual data output lines.

The arrangement shown in Figure 1 has disadvantages in that it slows down operation of the FPGA. The reason for this can be understood by reference to Figure 2 which is an equivalent circuit for the electrical path between the output line 38 of standard cell 10 and the input line 22 of standard cell 12. The resistor and capacitor inside dashed line 28 represent the parasitic components of pass transistor 28 which affect propagation on the line. The resistor 54 represents the channel resistance of the pass transistor 28, and is typically 1000 ohms. The capacitor 56 represents the parallel summation of the parasitic gate-to-source, gate-to-drain, gate-to-substrate and line-to-substrate capacitances of the lines to and from each pass transistor source, drain and gate terminal. The combination of series resistor 54 and shunt capacitor 56 represent a low pass RC filter that shunts high frequency components of the high speed data signal to the substrate and causes delay in the rise times of fast pulses in accordance with the RC time constant of the filter.

Loss of high frequency components caused by these RC low pass filters, and the RC delay factor tends to soften the edges of rectangular pulses and spread them out. This decreases the slope of the edges of high speed rectangular pulses thereby increasing rise

times. For example, consider high speed data input pulse 58 which will be assumed to be 10 nanoseconds in width. The pulse looks line pulse 60 after passing through the first low pass filter of pass transistor 28 and continues to degrade in both amplitude and pulse width as it propagates through each pass transistor, as represented by pulses 62 and 64.

Pass transistor interconnect suffer from the tradeoff between size and low turn-on resistance. Large size is not good because there are hundreds of thousands of pass transistors in the typical prior art gate array, and large size increases the die size and lowers yield and leaves less room for standard cells in the core array and interconnect lines. However, small size in a pass transistor, although it decreases its parasitic capacitance, also increases its parasitic turn-on resistance. Pass transistors also do not scale with technology. If a pass transistor is shrunk, its area and device capacitance are reduced, but its resistance stays constant because of voltage scaling. A third disadvantage of pass transistors is that the transistor of choice is a single n-channel device. Because of this, a V_T drop is encountered when a signal crosses a pass transistor. This starts to become a major disadvantage at technologies below 0.35 microns. As the supply voltage drops, there is less voltage across the pass transistor meaning less current. This causes designers to change their design goals to only maintain performance as opposed to improve it as linewidths shrink. In the prior art, designers have attempted to solve the pass transistor's problems using a hierarchy of lines that span different numbers of logic blocks in the core array. Other efforts have focussed on use of charge pumps to increase the gate voltage to lower the "on" resistance. Charge pumps become more difficult to implement at lower gate lengths because of the ever reducing thickness of gate oxides. These prior art attempts have made incremental improvements, but no great leap forward in performance.

Each MOS pass transistor such as transistors 28, 30, 32, 34, 36 and 50 operates in common gate mode. MOS devices operating in common gate mode exhibit no gain. Such passive devices are not coupled to any power supply so they supply no energy to replace the energy dissipated in the series resistance of the MOS device channel and the energy of the high frequency components siphoned off to the substrate through the parasitic shunt capacitances. This is the cause of the degradation of the amplitude and pulse width of the pulses 62 and 64.

Other FPGA companies use different forms of passive links called antifuse technology. A typical antifuse structure cross-sectional diagram is shown in Figure 3. In an antifuse FPGA structure, a first plurality of conductors are formed on a first conductive layer, one line of which is represented by metal layer 66. A second plurality

of conductors are formed on a second conductive layer which is insulated from the first conductive layer. This second plurality of lines is represented by metal layer 68. At the intersection of each line on the first layer with a line of the second layer, an antifuse structure like that shown in Figure 3 is formed. In each antifuse structure, the two conductive layers 66 and 68 are separated from each other by an insulating layer 70 which is typically polysilicon. The polysilicon layer has a gap etched through it to form a via from the metal layer 66 to the metal layer 68. Metal from layer 66 fills in this via. The via has a layer of nitride 72 formed therein to separate and insulate the metal of layer 66 from the metal of layer 68.

To make a connection between metal line 66 and metal line 68, a 10 volt programming signal is applied to metal line 66 relative to metal line 68. This voltage exceeds the breakdown voltage of the nitride layer 72 and causes punch through by arcing wherein molten metal from layer 66 fills a hole in nitride layer 72 caused by the arc and makes electrical contact with metal layer 68. This punch through metal has a parasitic series resistance associated with it and the lines themselves have parasitic capacitance to the substrate.

The equivalent circuit for the structure of Figure 3 is as shown in Figure 4. Again, the equivalent circuit is comprised of a series resistance 74, which is typically 300 ohms, and a shunt capacitance 76 which is comprised of the parallel combination of the parasitic capacitance of the fuse plus the parasitic capacitance of the lines to and from the fuse. This RC combination again defines a low pass filter which siphons off high frequency components of the high speed data to ground.

The antifuse technology suffers from the same deficiency of the passive MOS pass transistor in that the successive RC delay factors and loss of high frequency components to ground through the shunt parasitic capacitances rounds off the corners of high speed rectangular data pulses, spreads out the width of the pulses and increases the rise time.

Although the antifuse technology has less parasitic resistance and shunt capacitance than a passive, common gate MOS pass transistor, it also suffers from the additional infirmity of not being scaleable down to today's extremely small design rules. When an antifuse FPGA structure is scaled down to today's 0.35 micron design rules, the structures on the FPGA get so small that they do not have sufficient breakdown voltage protection to withstand voltages greater than 3 volts. This means that application of a 10 volt programming voltage to a programming line can cause arcing and punch through at places where not arcing or punch through is intended thereby causing shorts or destroying devices or both.

Active devices such as MOS and bipolar transistors and other active devices are much more easily scaleable than antifuse technology, so active devices capable of amplification are used for the switches of active links according to the teachings of the invention.

Referring to Figure 5, there is shown a diagram of a FPGA according to the teachings of the invention using active link technology which is more readily scaleable and which does not degrade the speed of the chip as much as passive links. In the symbolic example of Figure 5, two standard cells, logic gates or other circuitry 75 and 77 need to have a conductive path formed therebetween in order to implement some function desired by the user. When the integrated FPGA circuit is manufactured, it is manufactured with a plurality of standard cells of various designs, a plurality of logic gates, registers, multiplexers, ALUs etc. thereon. The boilerplate circuits can be any collection of known circuits and some FPGA chips intended for use in a particular field may have a different collection of boilerplate circuits than other FPGA circuits intended for use in another field of use. In the claims, these standard "boilerplate" circuits are referred to as standard cells, logic gates or other circuitry.

The FPGA chip after manufacture also has formed thereon a connection matrix, crossbar switch or some other form or arrangement of conductors and active links which is coupled to the boilerplate circuits and which can be programmed to make the desired connections between the boilerplate circuits to implement the desired functionality. In Figure 5, the connection matrix is represented by conductors 79, 81, 83, 85, 87 and 89 and the associated active links represented by boxes with the letter A inside them. The boxes with the letter M inside them represent memory cells in which are stored bits which define whether the particular active link presents a high impedance path between the circuit nodes between which is connected or a low impedance path. The term active link in the claims means both the switching circuit represented by the boxes with the letter A inside as well as the memory cell or other circuitry by which the switching circuitry is either turned off or turned on at the choice of the user to make the necessary connections to implement the desired functionality. Typically, the control signals needed to control the switching circuits of the active links are supplied by memory cells which are loaded with logic bits by means of a shift register (not shown) of which each memory cell typically is a part and which snakes throughout the FPGA chip. To program the FPGA to any particular functional configuration, this shift register is loaded with thousands of bits in a particular sequence so all the desired connections are made in the connection matrix.

The exact nature of the switching circuits of the active links can vary, but all the different embodiments for active link switching circuits share the common characteristic of having a gain of approximately one or more so as to not degrade the speed of operation of the FPGA when a signal traverses a long chain of active links. One way of looking at the active links is that they have the capability of replacing energy lost to the substrate through parasitics from the signal propagating through a long chain of active links. Of course, a competitor might conclude that if he makes an active link with a gain of 0.9, he could escape infringement while still enjoying a speed advantage over the prior art passive link FPGAs. The phrase "having a gain of approximately one or more" is not intended to limit the gain of an active link according to the teachings of the invention to exactly one or higher. Instead this phrase is intended to indicate that an active link according to the teachings of the invention will have sufficient gain to not seriously degrade the rise times of signals propagating of a long chain of, for example, 10 active links to the extent such rise times are degraded in propagating through a chain of, for example, 10 passive links of the prior art.

A connection matrix according to the teachings of the invention is characterized by a plurality of conductors which are electrically insulated from each other such as conductors 79, 81, 83, 85 , 87 and 89 in Figure 5. A plurality of programmable electrical connections between selectable combinations of these conductors is implemented using a plurality of active links. In the example of Figure 5, a programmable, easily reconfigurable data path between standard cell 75 and standard cell 77 is implemented by properly programming active links 91, 93, 95 and 97. Active link 91 is programmed by writing an appropriate bit into its memory cell to causing the switching circuit to form a low impedance path between conductors 81 and 79. Likewise, active link 93 connects conductors 87 and 83 and active link 95 connects conductors 83 and 89 and active link 97 connects conductor 89 and standard cell 77.

A process for forming a field programmable gate array circuit similar to that shown in Figure 5 comprises forming the boilerplate circuitry on the die using any integrated circuit fabrication process compatible with the technology in which the boilerplate circuits are to be implemented. Then a connection matrix of conductors is formed using the same process such that the conductors are electrically insulated from each other. Then a plurality of active links are formed at locations so as to be able to make programmable connections between at least some of the conductors in the matrix and between input and output nodes of the boilerplate circuits and at least some of the conductors of the connection matrix.

An important feature of the invention is therefore to guide signals along data paths of a field programmable gate array in such a manner so as to prevent degradation of the rise or fall times of the signal to the same degree as the degradation of these rise or fall times should the same signal propagate through a like number of passive links in a prior art field programmable gate array. This is done by applying sufficient gain at each active link to prevent degradation in the rise or fall times of the signal comparable to that which would occur if the same signal were to propagate through the same number of antifuse or common gate type MOS device passive links such as are found in the prior art field programmable gate arrays.

Figures 6A through 6D are examples of MOS inverter circuits, and Figure 10 is an example of a CMOS AND gate that could be used to implement the switching circuit of an active link according to the teachings of the invention. Figure 7 is an example of an NMOS inverter with a push-pull output stage and tristate circuitry that also could be used to implement an active link according to the teachings of the invention. The examples of Figures 6A through 6D were taken from the treatise Ong, *Modern MOS Technology, Processes, Devices and Design,* Section 9-2, pp. 189-197 (McGraw Hill 1984) ISBN 0 07-047709-4, the entire contents of which is hereby incorporated by reference. In the particular example of Figure 6A, an MOS inverter transistor 82 has as a load a saturated MOS device and is set up to apply a gain of approximately one or greater to the input signal. Lines 78 and 80 are the conductors in the connection matrix to be connected through the active link. The connection is controlled in programmable fashion by the logic level on an enable signal EN on line 86 coupled to the gate of enabling MOS transistor 88. The enable signal EN comes from the associated memory cell containing the programming bit or from another source of enable signals that defines the desired functionality of the chip. The high rail voltage defining a logic 1 level is on line 90, and the low rail voltage defining a logic 0 level is on line 92. Thus, even if the gain of the inverter is greater than one, the logic level of the signal output from a chain of such active links can never rise higher than the logic one level defined by the voltage on the high rail 90.

Figure 6B is an example of an MOS inverter active link using a passive resistor load 94, and enabled by an enable signal EN on line 86 coupled to MOS device 88. All the considerations mentioned above with regard to the switching circuit of Figure 6A apply equally to the switching circuits of Figures 6B, 6C and 6D, so only the differences in the switching circuits will be described below.

Figure 6C is an example of an MOS inverter active link using a nonsaturated MOS device 96 for a load operating in the triode region. In this embodiment, the enable signal

EN is a voltage higher than the high rail voltage on line 90 when the device is to be enabled so as to force MOS device 96 into the triode region. EN is low so as to cut off transistor 96 when the device is to be disabled. Any circuitry to convert the logic one level of the enable signal from the memory cell to the appropriate voltage to place the inverter in the triode region will suffice to practice this embodiment according to the teachings of the invention.

Figure 6D is an example of an MOS inverter active link using a depletion mode MOS device 100 as a load. As is the case for the embodiments of Figures 6A, 6B and 6C, the inverter is enabled and disabled in a programmable fashion by application of an enable signal EN on line 86 from an associated memory cell or other source to an MOS device 88. In one logic state of EN, the MOS device 88 couples the source of the transistor 82 to the low rail 92 to enable the device. In the other state of EN, transistor 88 is cutoff thereby disabling the device by decoupling the source of transistor 82 from the low rail.

All of the MOS inverter embodiments for active links in Figures 6A through 6D, when disabled, present a degree of isolation between the input line 78 and the output line 80 which may be sufficient in some applications and insufficient in others. When the inverters are disabled, the input line 78 is coupled to the output line 80 by the gate-to-drain parasitic capacitance of the inverter transistor 82. This small capacitance should provide sufficient isolation for all but the highest frequency data signals. However, high speed data signals having a frequency so high that inadequate isolation is provided by this gate-to-drain parasitic capacitance are not likely to be used with the MOS inverter active links shown in Figures 6A through 6D. This is because these active links directly couple MOS devices to the high speed signal path thereby slowing the signal down because of the effects of the parasitic components inherent in each MOS device. With high speed data, the biCMOS differential current mode logic active links disclosed later herein are preferred, because the MOS devices in those active links are not coupled to the high speed signal path. Therefore, the MOS inverters of Figures 6A through 6D are likely to provide adequate isolation when disabled for low speed data FPGAs.

If a higher degree of isolation is desired for low speed or some higher speed FPGAs, the active links can simply be bipolar, CMOS or MOS tristate buffers of the type which are well known in the art. An active link of the tristate buffer type will have its data input coupled to conductor 78 and its data output coupled to line 80. The enable signal on line 86 will be coupled to the tristate control input of the tristate buffer. When the enable signal is in a logic state to disconnect line 78 from line 80, the tristate

buffer will be put in a tristate mode wherein line 80 sees an essentially infinite impedance.

Figure 7 is an example of a noninverting MOS tristate buffer taught in *Modern MOS Technology*, pp 217-8 which is used as an active link in low speed FPGAs according to the teachings of the invention. An NMOS transistor 102 forms a basic MOS inverter with depletion mode load transistor 104. This inverter is coupled through tristate circuitry to a totem pole push-pull driver comprised of NMOS transistors 106 and 108. The reason for the totem pole output driver is that use of a simple inverter to drive output loads which are usually capacitive will result in uneven rise and falltimes. This can possibly be remedied by increasing the load transistor size to improve the risetime, but this inordinately increases DC power consumption and the area of the inverter. The push-pull driver of Figure 7 remedies this problem because when the output pull-down transistor 108 is turning on, the pull-up device 106 is simultaneously turning off. This ratioless configuration allows individual tailoring of device sizes for symmetrical waveform preservation.

Tristate operation can be achieved for the active link of Figure 7 by raising the enable signal EN on line 86 to a logic 1. In this state, output lead 80 is isolated from both the high and low rails 90 and 92 by high impedances. This high impedance state is achieved when both output devices 106 and 108 are turned off by activation of the enable signal EN. When EN is activated, tristate NMOS transistors 110 and 112 turn on thereby grounding the gates of output transistors 106 and 108. Simultaneously, transistors 114 and 116 are turned off by EN* (the complement of EN) going to logic zero. This cuts off the gates of transistors 106 and 108 from their connections to the input line 78 and the output node 117 of the inverter, respectively.

One of the simplest forms an active link according to the teachings of the invention can take is a NAND gate of Figure 8. This gate has A and B inputs and a C output. To use a NAND gate as an active link, the A input is connected to one of the conductors of the connection matrix to be connected and the C output is connected to another to be connected. The B input is connected to receive the enable signal which controls in programmable fashion whether the connection is or is not made. The truth table of Figure 9 shows how A and C are selectively coupled under the influence of the enable signal B.

Of course NAND gates come in many different implementations some of which may be passive. Any NAND gate implementation capable of connecting two or more lines together in programmable fashion and having a gain of at least approximately one (or enough to provide a significant improvement in degradation of rise and fall times

characteristic of propagation through chains of passive links) will suffice to practice the invention. The circuit of Figure 10 is one implementation of a NAND gate which can be used to implement an active link.

In the Figure 10 CMOS implementation of a NAND gate to practice the invention, PMOS transistors 120 and 122 are connected in parallel and have their gates coupled to receive the A and B signals, respectively. Two complementary NMOS transistors 124 and 126 connected in series between common source node 128 and ground or the low voltage source also have their gates coupled to receive the A and B signals. When B is high, the connection between A and C is enabled, and transistor 126 is turned on and transistor 122 is turned off. Thus, when A goes high, transistor 124 turns on and transistor 120 turns off thereby grounding the common source node 128 which is coupled to output C. When A goes low, transistor 120 turns on and transistor 124 turns off thereby coupling output C to the high voltage reference 130 making output C a logic high. Because the NAND gate of Figure 10 is an inverting gate, an even number of gates must be used to make the connections between standard cells to be connected.

In applications where more isolation is needed between the A and C lines than the CMOS devices provide, a tristate buffer can be added immediately after the switch.

Referring to Figure 11, there is shown another circuit diagram for a preferred form of active link which can make buffered connections between two lines in a connection matrix. An advantage of the active link of Figure 11 is that the deleterious effects of the series parasitic resistance and shunt parasitic capacitances of passive links in the prior art are reduced or eliminated and gain of approximately one or more can be applied to high speed signals propagating through the active link. This reduces the deterioration of rise and fall times of high speed signals as they propagate through a series of active links.

In the active link of Figure 11, three different scenarios can be implemented depending upon the state of the enabling signals. They are:

1. line 150 isolated from line 152;
2. line 150 drives line 152;
3. line 152 drives line 150.

Which of these scenarios is implemented depends upon the programming of the enable signals transmitted into the circuitry inside box 153.

In case 1, NPN transistors 154 and 156 coupled as emitter followers are both disabled by programmable switches in the form of PMOS devices 158 and 160. All the MOS enabling transistors in Figure 11 are referred to in the claims as programmable switches, but other programmable switching devices could be substituted. The PMOS

devices are both turned off by virtue of the enable signals EN1 and EN2 both being high. In addition, current-mode, common emitter node current switches 162 and 164 are both disabled by virtue of the enable signals EN3 and EN4 being low. This causes NMOS devices 166 and 168 to be turned off thereby disconnecting the common emitter nodes 170 and 172 from the low rail voltage supply 174. No power is consumed in this state, and lines 150 and 152 are isolated and undriven.

The low rail conductor 174 and a high rail conductor 201 are coupled to a voltage source for development of a potential difference therebetween.

In case 2, the current switch 162 is enabled by driving enable signal EN3 high thereby turning on NMOS device 166 to connect common emitter node 170 to the low rail 174. Current switch 164 remains disabled by virtue of enable signal EN4 remaining in a logic 0 state thereby pinching off the channel in NMOS device 168. Emitter follower 154 is turned off because PMOS device 158 is off by virtue of enable signal EN1 being high. In this state, if the voltage of the input signal on line 150 is above the voltage of the reference voltage REF at node 180, then emitter follower 156 will drive output line 152 in accordance with the signal on line 150 by virtue of being coupled to the output node 182 of current switch 162 through PMOS device 160. In order for this to happen, the enable signal EN2 is driven low thereby turning on PMOS device 160, and enable signal EN5, and enable signal EN5 is driven high so as to turn on NMOS device 184 so as to enable the emitter follower 156 by connecting line 152 to ground. Enable signal EN6 is low in this mode thereby pinching off the channel of NMOS device 186. The structure of NMOS devices 186 and 184 and 166 and 168 is such that when their enable signals are high, the on resistance of the device supplies sufficient emitter feedback for their respective current switches or emitter followers to work. The on resistance of devices 186 and 184 should however be low enough that the RC time constant of this on resistance coupled to the parasitic capacitance of lines 150 or 152 as the case may be is sufficiently low so as to not substantially degrade the rise time of the high speed data signals propagating through the active link.

If the input signal at line 150 is below the reference voltage REF at node 180 in case 2 operation, then current flow through the emitter follower 156 and NMOS device 184 pulls line 152 down to a logic zero level.

Operation in case 3 is similar but reversed to operation in case 2. In this case, line 152 drives line 150 through emitter follower 154. In case 3, the enable signal EN4 is high, and enable signal EN5 is low. Enable signal EN6 is high so as to enable emitter follower 154, and enable signal EN1 is low so as to turn on PMOS device 158. Enable signal EN2 is high so as to turn off PMOS device 160 thereby disconnecting

emitter follower 156 from current switch 162. In this state, when the signal on line 152 is higher than the reference voltage REF (set between logic 1 and logic 0 levels), emitter follower 154 drives output line 150 high. When the signal on line 152 is below the voltage of the reference voltage REF, emitter follower 154 pulls line 150 low.

The active link switching circuit of Figure 11 has 6 different enable signals the logic states of which control in which of the three states the active link operates. These enable signals EN1 through EN6 may be supplied from 6 separate memory cells like memory cell 200 in Figure 5, or they may be supplied by any other means which allows programmability of the FPGA active links to operate as desired.

An advantage of the active link of Figure 11 is that it does not load whatever driving device is coupled to it, and this is true regardless of whether line 152 is driving line 150 or line 150 is driving line 152. In either mode, the driving device sees the essentially infinite input impedance of an emitter follower buffer. In the mode where line 150 is driving line 152, the driving device sees the high input impedance of emitter follower 156. In the mode where line 152 is driving line 150, the driving device sees the high input impedance of emitter follower 154.

A useful FPGA architecture using the active link teachings of the invention is shown in Figure 12. In this symbolic arrangement, programmable or nonprogrammable logic elements 202, 204, 206, 208, 210, 212 and 214 are programmably connected to each other and other devices elsewhere on the integrated circuit die. These connections are made by active links designated by the circles and boxes with Xs inside them. The active links shown as Xs inside circles inside PLE 208 are short range active links suitable for driving short lines coupling a PLE to it nearest and next nearest neighbors. Examples of the types of active link circuits disclosed herein which could be used for such short range active links are any of the active link circuits disclosed herein with short line drive capability such as the NMOS inverters, NAND gates etc.

To account for situations where signals must connect to devices that are not neighbors, the active links symbolized by Xs inside boxes are provided. These active links are designed to drive longer lines to enable connection of signals between circuits separated by as much as the entire width or length of the integrated die. An example of good long range active link designs are the NMOS inverter with totem pole, push-pull driver stage shown in Figure 7 or the programmably bidirectional active link with emitter follower output stages of Figure 11.

The significance of using two different types of active links on an FPGA is that each different type of active link will have different parasitics, different delay characteristics, different power requirements and different input and output impedances

and different complexity in terms of the number of devices needed, the number of programming bits needed and the amount of chip area consumed. Some of these active links will be better than others for driving short lines and some will be better for driving long lines. Since most FPGA layouts have standard cells or logic circuits connected to neighboring circuits, it is beneficial to have at least two different types of active links and use the short range ones for most link sites and only use the bigger, more complex active links where absolutely necessary to drive long lines.

This allows high speed signals to propagate at much higher speeds than in the prior art. The reason is that the gain in the active link and/or the energy supplied to the propagating signal by each active link as it passes through the active link replenishes energy dissipated in parasitic resistance or lost to the substrate through shunt parasitic capacitance thereby arresting or reducing the amount of deterioration of signal rise times as a signal propagates across the chip. The circuit of Figure 7 is well adapted for use as the long range active links because of the totem poll, push-pull output stage.

Referring to Figure 13, there is shown a circuit diagram for a one-input-two-output multiplexer embodiment of an active link suitable for use in implementing programmable logic circuitry that must process very high speed signals. In the embodiment shown in Figure 13, emitter-coupled-logic (ECL) circuitry is used in the data path, and CMOS circuitry is used for steering the input signals from the inputs to one or more of the outputs. In other embodiments, other fast technologies may be used in the data path such as bipolar, Josephson junction, ballistic effect devices etc. The data inputs for high speed data are shown at A and A-. These two inputs are coupled to two ECL differential pairs comprised of a first pair of transistors E1 and E2 and a second pair of transistors E3 and E4. Transistors E1 and E2 have load resistors R1 and R2, respectively. Transistors E3 and E4 have load resistors R3 and R4, respectively. The first data input A is coupled to the bases of ECL transistors E1 and E3. The complementary data input A- is coupled to the bases of ECL transistors E2 and E4. The collectors of the E1 and E2 pair are coupled to the Y1 and Y1- outputs, respectively. The collectors of the E3 and E4 pair are coupled to the Y2 and Y2- outputs, respectively. Each of the ECL transistors is coupled to the high voltage supply rail 210 via a collector load resistor where R1 is the load resistor for transistor E1 and R4 is the load resistor for E4 etc.

The emitters of transistors E1 and E2 are coupled so as to share a common constant emitter current regulated by current source transistor CS1, and the emitters of transistors E3 and E4 are coupled so as to share a common constant emitter current regulated by current source transistor CS2. The bases of these two current source

transistors are coupled to a reference voltage VREF, and the emitters are coupled to the low voltage supply 212 through emitter feedback resistors 214 and 216 and through NMOS steering transistors N1 and N2. The gate terminals of transistors N1 and N2 are coupled to enable signal lines EN1 and EN2, respectively. These two enable signal lines are also coupled to two PMOS steering transistors P1 and P2, respectively, which are coupled between the high voltage supply 210 and the common emitter nodes 218 and 220.

The operation of the circuit of Figure 13 is as follows. The fundamental purpose of the circuit is to steer the signals in the data path on complementary signal lines A and A- onto one or both of the complementary output signal pairs Y1/Y1- or Y2/Y2- without substantially degrading the rise or fall times of the high speed signal so as to implement a one-input-two-output multiplexer type active link. This is done using the EN1 and EN2 enable or steering signals from associated memory cells and the CMOS steering transistor pairs N1/P1 and N2/P2, the CMOS steering transistors being outside the data path so as to not expose the high speed data to the effects of the parasitic resistance and capacitance of the CMOS devices. Those skilled in the art will appreciate that the concept illustrated in the circuit of Figure 13 can be extended to more outputs than two and can be reversed to steer input signals on one of a plurality of inputs onto a single output.

Assuming now for illustration that the EN1 enable signal is active, i.e., high. This causes the steering transistor N1 to conduct thereby connecting the steering transistor N1 to the low voltage power supply and causes the steering transistor P1 to be nonconductive. This has the effect of activating the output pair Y1 and Y1- by enabling the current source CS1 to draw the fixed current represented by arrow 224 out of node 218. Current source transistor CS1 stabilizes the current flow out of node 224 by virtue of the negative feedback to the emitter caused by emitter resistor 214 as is well-known in the art. Because transistor P1 is nonconductive, the common emitter node 218 is not held at the voltage of the high voltage supply 210, and the ECL transistors E1 and E2 are free to drive the outputs Y1 and Y1- as an ordinary current mode logic buffer/inverter under the influence of whatever data signals are present on the data inputs A and A- as in normal ECL differential mode operation. If the EN2 steering signal is not active, i.e., low, simultaneously with the active high state of the EN1 steering signal, the Y2 and Y2- outputs are deactivated. This results from the fact that the steering transistor N2 is not conductive thereby disabling the current source transistor CS2 from drawing current from common emitter node 220. When steering signal EN2 is low, the PMOS steering transistor P2 is conductive thereby driving the

common emitter node 220 to the voltage of the high voltage source 210. This affirmatively reverse biases the base-emitter junctions of the ECL transistors E3 and E4 rather than leaving the common emitter node 220 floating so as to positively cut off the E3 and E4 transistors and prevent any signal leakage from the inputs A and A- to the outputs Y2 and Y2.

In some embodiments where this positive cutoff of the ECL transistors of the pair associated with whatever steering signal EN1 or EN2 is low, is not necessary and a floating common emitter node 218 or 220 provide adequate isolation between the input and output when the associated current source is not active, the PMOS transistors P1 and P2 can be eliminated.

If the enable signal EN2 is high, the NMOS steering transistor N2 is conductive and the PMOS steering transistor P2 is rendered nonconductive. This has the effect of activating the two outputs Y2 and Y2- by connecting the current source transistor CS2 to the low voltage supply thereby causing the transistors E3 and E4 to drive the outputs Y2 and Y2- in accordance with whatever data signals are on the A and A-data inputs. This is true regardless of whether steering signal EN1 is simultaneously active high. If EN1 is simultaneously low when EN2 is high, steering transistor N1 is nonconductive and steering transistor P1 is conductive. This drives common emitter node 218 to the voltage of the high voltage rail and reverse biases the emitter-base junctions of the ECL transistors E1 and E2 thereby isolating the inputs A and A- from the outputs Y1 and Y1-

By controlling which of steering signals EN1 and/or EN2 are high, it is possible to connect the input signal pair A, A- to either or both of the output signal pairs Y1, Y1- or Y2, Y2-. Thus, by control of the states of signals EN1 and EN2, it is possible to electronically control the switching of very high speed signals at an input to any of a plurality of outputs without substantially slowing down the signals even though CMOS steering transistors are used. Because the resistivity of the NMOS steering transistors N1 and N2 is much lower than the resistance of the emitter feedback resistors 214 and 216, the presence of the NMOS transistors N1 and N2 in the path between the emitters of the current source transistors CS1 and CS2 to the low voltage rail does not appreciably affect the speed of operation of the circuit.

The structure of Figure 13 can be extended to more ECL differential pairs driving more output pairs, but there is a limit imposed by the loading on the input signal lines A and A- caused by the junction capacitances of the ECL transistor bases. Also, since the beta factor of the ECL pairs is not infinite, adding more ECL pairs causes the base current to exceed accepted ECL limits of no more than 10-20 bases coupled to one signal.

The preferred limit of the number of bases which can be coupled to input signal lines A and A- is from 4 to 8. It is preferable for the load on A and A- signal lines to not be dependent upon the number of transistors connected thereto.

Referring to Figure 14, there is shown a circuit diagram for a two-input-single-output multiplexer active link for steering high speed signals without substantially degrading the rise or fall times thereof. A first ECL transistor pair E5 and E6 share a common emitter node 230, while a second ECL transistor pair E7 and E8 share a common emitter node 232. Common emitter node 230 is coupled to a constant current source comprised of transistor CS3 and emitter feedback resistor 234. This constant current source is turned off and on by an NMOS steering transistor N3 which couples the current source to the low voltage supply line 236. The transistor pair E5 and E6 each have a load resistor, R5 and R6, respectively, which is shared with a second ECL transistor pair E7 and E8 via a pair of single output lines Y and Y-. The second transistor pair E7 and E8 share emitter node 232 and share a constant current source comprised of transistor CS4 and emitter feedback resistor 238. The CS4 constant current source is selectively coupled to the low voltage supply rail 236 by an NMOS steering transistor N4. As in the case of the embodiment of Figure 13, two PMOS steering transistors P3 and P4 are used to positively control the voltage of common emitter nodes 230 and 232, respectively.

The first ECL transistor pair E5 and E6 is enabled when the enabling signal EN1 is high. This condition turns the NMOS transistor N3 on and couples the current source transistor CS3 to the low voltage supply 236. The transistors E5 and E6 are coupled to a high voltage supply line 240 through their respective load resistors R5 and R6. When EN1 is high, PMOS transistor P3 is off which releases the common emitter node 230. Thus, transistors E5 and E6 are enabled to drive the output lines Y and Y- under the influence of whatever signals are on the high speed input signal lines A and A-. Note that if EN1 is high, care must be taken to insure that EN2 is not simultaneously high as this would cause a conflict in that ECL pair E7 and E8 would be simultaneously trying to drive the output lines Y and Y- at the same time transistors E5 and E6 were trying to drive the same lines, possibly with conflicting signal levels. This conflict is avoided if the steering signal EN2 is low when steering signal EN1 is high, because a low EN2 causes PMOS transistor P4 to be turned on which drives the shared emitter node 232 to the voltage of the high voltage supply line 240. This disables E7 and E8 by reverse biasing the emitter-base junctions thereof. Likewise, when EN1 is low, transistors E5 and E6 are disabled in the same way.

In alternative embodiments of the circuit of Figure 14, interlock circuitry is employed to prevent both EN1 and EN2 from being active high simultaneously. Also, in some embodiments, the PMOS transistors P3 and P4 can be omitted where leaving the shared emitter node floating is an acceptable way of disabling the ECL transistor pairs.

The multiple input, single output arrangement of Figure 14 can be extended to many different input pairs driving many different ECL pairs sharing a single output pair, as will be apparent to those skilled in the art. If such a circuit were to be implemented as an integrated circuit, all transistors whose collectors are connected to the same output line could share the same collector tub on the integrated circuit die thereby creating vast savings in layout area. Thus, for example, four separate input pairs could drive four ECL transistor pairs sharing a single output pair and a single pair of load (pull up) resistors. The four transistors coupled to one output line of the output pair would share the same collector tub and likewise for the four transistors coupled to the other output line. At most one of the four ECL pairs would be enabled by its corresponding steering signal while all other steering signals would be inactive.

Such an embodiment is shown symbolically in Figure 15. In the notation used in Figure 15, ECL pair E5 and E6 with pull up resistors R5 and R6 and their associated current sources and MOS steering transistors are represented by switch 250 while ECL pairs without pull-up resistors such as transistors E7 and E8 and their associated current sources and MOS steering transistors are represented by switches 252, 254 and 256.

In Figure 15, the first ECL transistor pair is driven by high speed signal input lines A and A- in the data path, while the second ECL transistor pair is driven by high speed input signal lines B and B-. Both ECL transistor pairs drive a single pair of shared output signal lines Y and Y- and share a single pair of pull up resistors.

Referring to Figure 16, there is shown a symbolic diagram of a four-input-three-output crossbar switch comprised of active links of either the long range or short range variety. Each triangle in Figure 16 represents an active link. The particular cross-bar switch shown has three modules like that shown in Figure 15 interconnected such that the data inputs of the first module comprised of switches 258, 260, 262 and 264 also drive the data inputs from corresponding switches in the other two modules. Specifically, the A and A- data inputs to switch 258 are coupled not only to the data inputs of switch 258, but also to the data inputs of switches 266 and 268 via lines 259 and 261, and the B and B- data inputs drive the data inputs of both the switch 260 and the switches 268 and 272 via lines 263 and 265. The C and C- data inputs are similarly connected so as to drive the data inputs of switches 262, 274 and 276 via

lines 267 and 269, and the D and D- data inputs are coupled to drive the data inputs of switches 264, 278 and 280. For clarity of the figure, the separate enable inputs of each switch are not shown, but each switch has an enable input coupled to receive a steering signal such as the signal EN1 in Figures 13 or 14. These steering signals are coupled to the CMOS steering transistors that control enabling of the ECL transistor pair of each switch in the manner described above for the circuits of Figures 13 and 14. The concept of Figure 16 can be extended to any number of modules and any number of inputs and outputs limited only by the fan-out specifications or loading limitations on each input and output of the switch.

Operation of the crossbar switch of Figure 16 is a straightforward function of activating selected ones of the steering signals. For example, if it is desired to pass the D and D- signals on to only the W and W- outputs, the enable signal to switch 264 would be activated and all other enable signals to all other switches would be inactive. If the D and D- outputs were to be steered to the X and X- outputs, the enable signal for only switch 278 would be active, and all other enable signals to all other switches would be inactive.

Any input pair can be coupled to any one or more output pairs in the architecture of Figure 16, and two or more inputs can be coupled to two or more outputs simultaneously as long as no output is coupled to more than one input at any particular time. For example, the A and A- inputs can be coupled to the W and W- and X and X- inputs simultaneously while the D and D- inputs are simultaneously coupled to the Y and Y- inputs. Many other combinations are also possible as will be apparent to those skilled in the art.

The architecture of the crossbar switch of Figure 16 can be extended to larger numbers of input pairs and/or output pairs.

Simulations of the operation of a 4 x 4 (four input pairs and four output pairs) crossbar switch having an architecture like that of the circuit of Figure 16 has shown propagation delays of about 1/4 nanosecond. This is much faster than the propagation delays of such crossbar switch circuits implemented using CMOS in the data path. That is, the time it takes for a change of level on any output pair to propagate through the circuit and cause a corresponding change in level on any one or more selected output pair is 1/4 nanosecond.

The circuits shown in Figures 13 through 16 all use current mode logic which has a maximum output voltage swing on the order of 300 millivolts. If the output voltage swing is stretched to a value more than 300 millivolts, soft saturation or total saturation can occur in the ECL transistors. This is highly undesirable because

saturation or soft saturation of current mode logic switches substantially decreases the switching speed thereof. Thus, in the preferred embodiment, emitter followers are used as output buffers so as to increase the permissible output voltage swing.

There are other reasons to use emitter followers either as active links or in conjunction therewith. Specifically, in addition to having very high input impedance which is useful to prevent the driving circuit from being loaded down, emitter followers can be used to shift the voltage levels so as to drive other logic families. Emitter followers can also be used to create higher current source or sink capacity for driving long lines which is useful for construction of the long range active links of Figure 12.

To provide maximum flexibility, it is desirable to be able to couple the output of a current mode logic switch active link to any one or more of a number of emitter follower arrangements, some of which may have different characteristics such as different output voltage levels, logic swing or current source or sink capacity.

To provide this flexibility, the circuit of Figure 17 may be used alone as a multiplexing active link or in conjunction with other active link circuits to provide long line drive capability, voltage level shifting, better isolation etc. In the circuit of Figure 17, a data output line A from the output of a current mode logic switch or any other type of active link or boilerplate circuit is coupled via line 251 to two emitter followers comprised of ECL transistors E9 and E10 which drive output lines F1 and F2. Each of these emitter follower transistors has an associated current source and associated CMOS enabling circuitry. Specifically, transistor E9 drives output line F1 and has its emitter coupled to a current source transistor CS5 which has an emitter feedback resistor R7 and which has its base coupled to a constant reference voltage Vref as was the case with the current source transistors of the circuits of Figures 13-16. The CMOS enabling circuitry for emitter follower transistor E9 is comprised of NMOS transistors N6 and N7 and PMOS transistor P6. Likewise, emitter follower transistor E10 has its emitter coupled to a current source transistor CS6 having emitter feedback resistor R8. The base of the current source transistor CS6 is coupled to the constant reference voltage line Vref. The enabling CMOS circuitry for the current source E10 is comprised of NMOS transistors N8 and N9 and PMOS transistor P7.

Emitter follower E9 is enabled when the steering signal EN9 is active high and complementary steering signal EN9- is active low. This state causes NMOS transistor N7 to be turned on thereby activating the current source transistor CS5 by coupling its emitter to the low voltage supply line 252. Because EN9- is active low, NMOS transistor N6 is turned off and PMOS transistor P6 is turned on thereby allowing the base of transistor E9 to assume whatever voltage high speed input signal A currently

has. Note that although a PMOS transistor P6 is in the high speed signal path, the load on this transistor is very light comprised of only one ECL transistor base and one NMOS transistor drain. This light load does not appreciably slow down signal propagation. It is necessary to use the P6 transistor in the embodiment shown in Figure 17 because it is necessary to disconnect the high speed signal A from the base of transistor E9 when the base is coupled to the low voltage supply 52 so that the high speed data signal is not loaded down thereby slowing signal propagation.

To disable emitter follower E9, steering signal EN9 is driven to its inactive low state and complementary steering signal EN9- is driven to its inactive high state. This state causes NMOS transistor N7 to turn off and NMOS transistor N6 to turn on simultaneously with PMOS transistor P6 turning off. This causes current source transistor CS5 to be disconnected from the low voltage supply line 252 and become inactive thereby disabling the emitter follower transistor E9. Simultaneously, the base 254 of NPN transistor E9 is coupled to the low voltage supply 252 to reverse bias the base-emitter junction and the base 54 is cut off from the A data input line 251 by virtue of transistor P6 turning off. By coupling the base 254 of transistor E9 to the low voltage supply, the base-emitter junction of NPN transistor is reverse biased thereby preventing any voltage source coupled to the F1 output from accidently turning transistor E9 on.

Emitter follower E10 works in the same fashion as emitter follower E9. However, it may have a different physical geometry or emitter feedback resistor R8 may have a different value so as to present different voltage levels on output line F2. Further, emitter follower E10 may be designated so as to be able to source more current to output line F2 to drive a long line. Thus, when steering signal EN10 is active high and steering signal EN10- is active low, NMOS transistor N9 is on enabling the current source transistor CS6 and NMOS transistor N8 is off while PMOS transistor P7 is on thereby connecting the base 256 of NPN transistor E10 to high speed data input A. To turn off E10, steering signal EN10 is made inactive low and steering signal EN10- is made inactive high.

Note that the architecture of the circuit of Figure 17 allows the high speed data signal A to drive either output F1 or output F2, or both simultaneously or neither depending upon the states of the steering signals EN9 and EN10 and their complements. Each emitter follower has a gain of approximately one or greater and presents a high input impedance to the data input A so as to prevent unnecessary loading of whatever circuit is driving data input A.

Another high speed data input, B, on line 251 is coupled to the base of an NPN emitter follower transistor E11 which also drives output F2. If level shifting of the output swing of output F2 were desired when driven by input B, emitter follower transistor E11 could be replaced by two transistors in series such that two base emitter drops of approximately 850 millivolts would be imposed between the high voltage supply line 60 and the output F2 when the emitter follower E11 is turned on. Emitter follower E11 and its steering circuitry works the same way as emitter followers E9 and E10. Specifically, when steering signal EN11 is active high and its complement EN11- is active low, NMOS transistor N10 is on and activates current source transistor CS7 by coupling the emitter thereof to low voltage supply line 252. The base of transistor CS7 is coupled to the constant reference voltage Vref as are the bases of current source transistors CS5 and CS6. Simultaneously, NMOS transistor N11 is turned off by the low state of EN11- and PMOS transistor P8 is turned on thereby connecting the high speed data input B to the base of emitter follower transistor E11. This causes the changes in logic level of high speed data input signal B to be reflected on output F2 while imposing the buffering, level shifting and boosted current drive benefits of the emitter follower E11 between the high speed data input signal B and the output signal F2 which follows it.

Thus, by driving steering signal EN11 active high and steering signal EN10 inactive low and the complementary steering signals to their corresponding active/inactive states, it is possible to drive output F2 with input B. Likewise, by driving steering signal EN10 active high and steering signal EN11 inactive low and the complementary steering signals to their corresponding active/inactive states, it is possible to drive output F2 with input A. It is not permitted to have both steering signals EN10 and EN11 active high at the same time, although it is permitted to have both inactive low simultaneously.

Note that the inputs A and B on lines 251 and 253 may be coupled to any of the outputs shown in Figure 13-16 such as Y or Y- etc., and note that duplicate emitter follower circuitry may be used to couple to the complementary outputs. Also, the outputs F1 and F2 may be coupled to the inputs of the single ended circuit of Figure 18 to provide any necessary one Vbe drop (base-emitter voltage drop) to properly bias that circuit. The number of possible permutations and combinations of the fast switching circuits and emitter follower circuitry that does not impede the switching speed according to the teachings of the invention are too numerous to draw them all, but they will be apparent to those skilled in the art. Any fast switching circuit that uses MOS enabling circuitry that is substantially removed from the data path to enable the switch

or do a steering function for the high speed data signals is equivalent to what is taught herein and intended to be within the scope of the claims appended hereto.

The consequence of use of the architecture of Figure 17 in conjunction with the architecture of any of Figures 13-16 is that the high speed switches of Figures 13-16 may be coupled to any other type of logic family regardless of the logic levels of the logic family to which the high speed switches are to be coupled. The level of the output signals at outputs F1 and F2 can be raised by raising the voltage at the high voltage supply line 260 or changing the values of emitter feedback resistors R7, R8 and R9 and changing the characteristics of the current source transistors and/or changing the reference voltage Vref to alter the level of current flowing through the emitter feedback resistors. Likewise, output level voltages can be shifted downward by coupling more emitter follower transistors in series so that all transistors in the chain turn on or off simultaneously with changes in the input signal levels and so as to impose their base-emitter voltage drops in series between the high voltage supply line 260 and the corresponding output. This level shifting can be done without loss of the advantage of ECL speeds. This provides great flexibility to designers. For example, in well-known differential mode cascade logic, one set of data inputs coupled to the lower differential pair substituted for the normal current source must be driven between level changes which are uniformly one base-emitter voltage drop (approximately 850 millivolts) below the corresponding levels of the other set of data inputs coupled to the upper differential pair. This can be done using the emitter follower technology of Figure 17, for example by driving one set of inputs with the F1 output and driving the other set of inputs coupled to the lower differential pair with output F2 and substituting a pair of series coupled emitter followers for single emitter followers E10 and E11.

Those skilled in the art will appreciate that the concepts illustrated in Figure 17 can be extended such that input A can drive more emitter followers and more outputs, and the crossbar switching capabilities of input A or B being able to drive input F2 can be extended to more inputs and outputs by modification of the circuit of Figure 17. Likewise, the concepts illustrated in Figures 13-16 may be extended to more inputs and more outputs and can be extended to single ended technology. Generally, differentially coupled current mode logic devices are preferred, because the logic swing can be reduced with adequate noise immunity and therefore great speeds can be achieved. However, where integrated circuit space is an issue, and the number of transistors is to be held down, single ended logic can also be used.

For example, a single-ended, fast OR gate employing the teachings of the invention is shown in Figure 18. In this circuit, a differentially coupled pair current

mode NPN transistors 280, 281 and 284 are coupled to share a common node 283. A reference signal Vbb is coupled to the base of transistor 281, and high speed data input signals A and B coupled to the bases of transistors 280 and 281 have logic states which swing both above and below the level of Vbb. A constant current source comprised of transistor 286 and resistor 288 selectively drives the common node 283 when NMOS transistor 290 is turned on. This occurs when steering signal EN12 is active high. When this occurs, PMOS transistor 282 turns off and common node 283 is free to seek whatever voltage it normally assumes when the differential pair is enabled. The transistors 284, 280 and 281 then drive the outputs Y and Y-. Pullup resistors 292 and 294 couple the outputs to the high voltage source. The voltage swings of the signals on inputs A and B can be increased to increase noise immunity, but the high level of either signal cannot be higher than one base-emitter voltage drop below the level of Vh on the line 300. Thus, an emitter follower according to the teachings of Figure 17 could be used to drive the A and B inputs if necessary to provide the necessary one base-emitter drop.

The concept of Figure 18 can be extended to any of the other switches or arrays shown in Figures 13-16 as will be apparent to those skilled in the art. Further, although NPN bipolar current mode logic technology is used for illustration, PNP bipolar technology could also be used, and any differentially coupled circuit could also be single ended. Further, other high speed switching technologies either now existing or to be invented in the future could also be used to implement the teachings of the invention if the slower enabling/steering logic is kept out of the high speed data path.

Referring to Figure 19, there is shown an overall top view of the layout of a typical FPGA employing RIUs according to the teachings of the invention and showing the relative positions thereof. Blocks 310 and 312 are typical logic blocks of the core array of programmable logic blocks which may be programmed to define the overall functionality of the chip. Blocks 314 and 316 are programmable I/O cells which are typical of the I/O cells which ring the array of logic blocks. The cross-hatched area 318 which rings the array of logic blocks and lies between the logic blocks and the ring of I/O cells is called the padframe interface circuit. This circuit is used to connect the logic blocks of the core array to the ring of I/O cells.

Typically, in the prior art, the padframe interface circuit area was very congested with circuit lines running all over and multiple metal layers in the integrated circuit structure. This complicated circuit took many hours to design and do the placement and routing. Each time a new chip in a family of FPGA was introduced with a bigger array of logic blocks, the padframe interface circuit had to be redesigned - a

process which considerably slowed down the time to introduction of the new chip to the market. The RIUs according to the teachings of the invention speed up that time to market by considerably shortening the time to design the padframe interface circuit.

Figure 20 is a block diagram of the functionality of a typical logic block such as logic block 310. The RIUs of the invention are not limited to working with the particular logic block of Figure 20 however. A typical logic block has an AND/OR section 320, a first multiplexer 322, an arithmetic section 324, a comparator section 326, a 32-bit RAM section 328, a second multiplexer section 330, a first flip-flop 332, a second flip flop 334, a combinatorial logic section 336 and a third multiplexer 338. Each logic block has 18 inputs shown at 340 and 3 outputs shown at 342. The three outputs are driven by the combinatorial logic or the flip flops. Input 344 is a local, global or quadrant clock signal, and input 346 is a local or global set/reset signal.

Each column of logic blocks is separated from the next column of logic blocks by a column of north-south connection lines shown symbolically as four lines, column 350 being typical. Each of the 4 north-south lines in each column represents 9 individual conductors for a total of 36 traces. Of these 36 conductors, 30 conductors are outputs and 6 are inputs. Any 24 of the output conductors are coupled to the RIUs in the padframe interface, the other 6 stopping at the edge of the array. All 6 of the input conductors are coupled to the RIU connection matrices, as will be described in more detail below.

The single lines 352 and 354 coming out of logic block 356 represent the collection of input and output lines to and from logic block 356. These input and output lines are coupled to selected ones of the 36 north-south column lines in column 350 using a not fully populated matrix of programmable connections represented schematically by the X-like symbols of which symbol 358 is typical. These programmable connections are, in this particular core configuration, implemented using one-way tri-state driver built in CMOS. The one way connections are from the east-west signal lines represented by line 352 to the north-south signal lines of column 350.

Figure 22A is a schematic diagram of the preferred tri-state driver, and Figure 22B is the truth table defining its operation. When the enable signal on line 360, which is defined by a programmable bit stored elsewhere on the chip in memory, is low, the output line Y is tri state and no transition on input A gets through to output Y. When enable signal E is high, logic levels and transitions on input A are inverted and appear on output Y. Note that the tristate driver is an active device which regenerates the input signal on A on output line Y thereby replenishing energy lost to parasitics from the signal on input A. The use of any of the active repeaters described herein for

interconnects has the advantage of speeding up the operation of the chip by segmenting the load. In the prior art pass transistor based interconnects, a logic block driving a signal to another logic block through a long net of pass transistor sees the entire load to the next logic block. When the same interconnect is implemented with active repeaters, the load is segmented such that each active repeater sees only the load to the next active repeater. The active repeater fully regenerates the signal and provides the optimum amount of drive to its load. Another significant advantage of active repeaters is fanout. With pass transistors, the V_t drop of each pass transistor reduces the voltage and current available to drive a load so if there is significant fanout to multiple loads, the problem gets even worse. With an active repeater, there is plenty of current to drive many loads, so performance is not significantly degraded as fanout increases within the current drive capability of the active repeater.

Other implementations for the tri-state driver in biCMOS or other technologies compatible with the logic block construction can also be used. Programmability of the connection is provided by a signal defined by a bit stored in memory and supplied on line 360 to make or block the connection. The matrix of connections can be fully populated if sufficient chip area exists and power consumption and cooling issues are adequately managed.

Bidirectional active repeaters, represented by blocks 362 and 364 in Figure 21, are used to provide programmable connections between the conductors represented by lines 366 and 368 and the programmable connection matrix at 370 such that signals can be driven in either direction. The symbols 362 and 364 represent a bidirectional active repeater on each of the individual signal lines represented by buses 366 and 368. Each of these bidirectional active repeaters preferably has the structure shown in Figure 23A and operates in accordance with the truth table shown in Figure 23B, but they can also have the structure of any of the other bidirectional repeaters disclosed herein.

The bidirectional active repeaters act as buffers and serve several functions. First, pairs of bidirectional active repeaters on the same line act as natural dividers to segment one conductive line into multiple segments. These active repeaters shield the line segments between the repeaters from the parasitic capacitances and resistances affecting the line outside the segment of interest. This segmentation causes the capacitive load seen by the drivers at the output of a logic block or an I/O cell to be fixed and limited to the amount of parasitic capacitance coupled to the particular segment to which the driver is connected. For example, conductive line 366 has two separate segments defined therein. The first segment lies between active repeater 364 and active

repeater 357. The second segment lies between active repeater 357 and active repeater 359. Focussing on the first line segment, suppose a driver in logic block 361 has its output coupled to one of the lines of bus 355 and suppose that line is coupled by a tri-state driver to line 366. In this situation, the parasitic capacitances seen by this driver coupled to the first segment of line 366 will be limited to only those parasitic capacitances of devices actually physically connected to this first line segment. Specifically, Figure 33 is an approximate equivalent circuit of the load parasitic capacitances seen at the output of a driver 363 within logic block 361 by virtue of it being coupled to the first segment of line 366 (represented by reference number 366' in Figure 33). The gate and drain parasitic capacitances of repeaters 364 and 357 are represented by capacitors 364' and 357'. The driver 363 also sees the capacitance 365' of the disabled line buffer 365' inside logic block 365 in Figure 21 since this driver cannot be simultaneously enabled with driver 363 to drive the same line segment. Driver 363 also sees the parasitic gate (unidirectional repeaters—370' and 371' are gate parasitics only, but for local bidirectional repeaters 370' and 371' represent both gate and diffusion parasitic capacitances) capacitances 370' and 371' representing all the gates of tri-state drivers connected to line 366' in the programmable vertical-to-horizontal connection matrices 370 and 371, respectively, at the intersection of the vertical line segment 366' and horizontal buses over which it passes. Finally, driver 363 also sees the parasitic capacitance between the line segment 366' itself and the substrate, represented by capacitor 373.

Because driver 363 sees a constant capacitive load regardless of whether the active repeaters are programmed to be "on" or "off", it can be optimally structured to drive this load. This is also true of all the active repeaters in the FPGA - they can all be designed optimally to drive a fixed capacitive load. Specifically, optimized active repeaters can be used for the repeaters that segment the individual signal conductors of the horizontal and vertical buses in the core array and which segment the I_lines and O_lines of the alternative RIU structure described herein and which populate the programmable connection matrices in both the core array at the intersections of the horizontal and vertical buses and elsewhere and at the intersections of the individual conductors of the vertical buses or columns with the I_lines and O_lines of the RIUs.

Note in Figure 21 that the placement of active repeaters in the individual conductive signal lines of the vertical buses is staggered in symbolic groups of four. Likewise, the active repeaters in the horizontal buses are also staggered in symbolic groups of three. This staggering of active repeaters improves access. "Access" is defined as the number of logic blocks to which the output of a single logic block can

connect while crossing no "repeater boundaries" (a repeater boundary is an active repeater dividing a line into two segments) and only one vertical-to-horizontal active repeater connection between a vertical bus conductor and a horizontal bus conductor). Having a high degree of access is important in an active repeater for highest performance and ease of routability. The staggered nature of the active repeaters improves access because the routing regions are not regular, and this irregularity allows the output of a logic block to access a much greater area. If the active repeaters were not staggered, any logic block output can only get to 13 other logic blocks (in the particular 2x2 core structure with an active repeater on every vertical and horizontal line at every other logic block) before crossing a repeater boundary. Crossing a repeater boundary is not a serious disadvantage but it does add another repeater delay and slows down performance slightly. With staggered repeaters in some embodiments within the genus of the core array configuration invention, a single logic block can access 8 other logic blocks not including itself without crossing a repeater boundary in either a vertical conductor or a horizontal conductor. In this case, the delays to all 34 logic blocks are equal. If crossing a single repeater boundary is allowed, the number of logic blocks a single logic block can access grows to a substantially larger number. Higher degrees of access greatly simplifies the placement and routing software designed to decide which programmable connections to make to implement a desired functionality and minimize interconnect delays. This also lowers the need to place timing critical circuits in the same region of the array.

Driver 363 in Figure 24 is only symbolic. Figure 25 represents the actual configuration of the output drivers and output lines from each logic block. Each logic block can output only three signals, labelled F, R and M. Each output drives ten tri-state drivers symbolized by drivers 375 and 377 in the case of output signal F. The ten output lines for each of these three sets of ten tri-state drivers intersect with the 30 output lines of the vertical channel, represented by vertical bus 350' representing only the 30 output lines of the vertical channel 350 and ignoring the six input lines. At preselected intersections, direct connections are made without the use of active repeaters.

Note that in the equivalent circuit of Figure 24, there are no parasitic capacitances representing loads coupled to line 366 above repeater 364 or below repeater 357. Also, note that there are no parasitic capacitances coupled to line segment 366' representing loads coupled to any of the lines of the horizontal buses passing through programmable connection matrices 370 and 371 since those are blocked by the

tri-state driver active repeaters in the programmable connection matrices 370 and 371.

This segmentation by active repeaters limits the capacitance on the segment of the line between two active repeaters because there is no "direct connection" between the line segments coupled to different terminals of the same active repeater. As the phrase is used in the claims, the lack of a "direct connection" means that there is no direct conductive path from one line segment to the other such that charge carriers from one line segment can flow directly through the active repeater to the other line segment. In a pass transistor FPGA, charge carriers can flow from one line segment coupled to the source to the other line segment coupled to the drain through the channel when the pass transistor is programmed to be "on". This direct path from one line segment to another does not exist in the preferred MOS inverter active repeater because in an MOS-based active repeater, one line segment is typically coupled to the gate of the MOS inverter and the other line segment is coupled to the drain thereof. There is no direct conductive path through the MOS inverter in which electrons can move from one line segment to the other. Although some current may flow between the line segments momentarily by capacitive coupling between the gate and the drain, after that parasitic capacitance is charged, no further direct current will flow between the line segments. In contrast, in a pass transistor, current originating from the driver of a logic block or I/O cell can flow directly from one line segment coupled to the source of the pass transistor to the other line segment coupled to the drain of the pass transistor through the channel of the device when it is turned on and start charging all the parasitic capacitances coupled to the drain of the pass transistor. This slows down propagation of a signal from this driver because propagation is measured by the time it takes from launch of a 0-to-1 logic transition at the output of the driver to attainment at the other end of the line of a predetermined voltage on the 0-to-1 transition. Charging of all the new parasitic capacitances added to the load of the driver as each new line segment is added in a pass transistor based FPGA as an interconnect is built by the routing software naturally slows down the propagation time of a logic transition from the output of the driver to the other end of the interconnect.

With an active repeater, regardless of whether it is programmed "on" or "off", there is no direct current path between the line segments for charging current to flow from a driver coupled to a first line segment to one or more parasitic capacitances coupled only to the second line segment. Thus, the additional propagation delay of adding each new line segment in a pass transistor based FPGA does not occur in an FPGA where the conductive lines are segmented by active repeaters.

Another significant advantage of this segmentation of lines by active repeaters is that it allows the same line to be used for multiple interconnects by using different segments of it for different interconnects.

However, the key advantage which arises out of the use of active repeaters, whether they are bidirectional or unidirectional tri-state drivers, is the segmentation of the capacitive loads seen by the drivers at the outputs of logic blocks and the resulting predictability of the delays which that entails. With the active repeaters segmenting the lines in either the columns or rows of the core array or in the I_line or O_lines of the RIUs, the capacitive load seen by the drivers at the output of the logic blocks or I/O cells does not change regardless of whether the active repeaters are programmed to be "on" or "off" because the parasitic capacitances coupled to line segments other than the line segment to which the driver is coupled are not being charged by current from that driver. This is true because the segments of a line connected by an active repeater are not directly connected so that current can flow from one segment directly to the other line segment. Although when the active repeater is programmed to be on, current/voltage in one line segment will cause current flow in the other line segment, the current flowing in the other line segment is current from the power supply and not from the output of the driver driving the first line segment. This is because, in the case of MOS inverters or bidirectional active repeaters, one segment of the line is coupled to a gate of an MOS device of the active repeater and the other segment is coupled to a drain of an MOS device. As a result, the only load seen by a driver coupled to drive the segment coupled to the gate is the gate capacitance of the active repeater plus any other parasitics coupled to that segment, and any parasitics coupled to the segment coupled to the drain are not seen by any driver coupled to the line segment coupled to the gate. Likewise, drivers coupled to the segment coupled to the drain see only the load of the drain plus any other parasitic coupled to that segment, and any parasitics coupled to the segment coupled to the gate of the active repeater do not affect the load of the segment coupled to the drain. This is true whether the active repeater is on or off because there is no direct connection between the segments. This situation is not true of a pass transistor based interconnect, because when the pass transistor is turned on, there is a direct current flow between the line segments and all the parasitics coupled to both segments coupled together through the pass transistor will affect any driver coupled to either segment.

This simple fact of direct connection in pass transistor interconnect FPGAs causes unpredictability of the loads affecting drivers at the data outputs of logic blocks or I/O cells in pass transistor based FPGAs and causes exponential rises in delays in

these prior art FPGAs as the length of an interconnect gets longer by connecting multiple segments together through multiple pass transistors.

The unpredictability of the delay in pass transistor based FPGAs also substantially complicates the routing software used to route interconnections in these FPGAs. Routing software attempts to minimize the delays involved in making a connection from one logic block to another or from a logic block to an I/O cell. As pass transistors are programmed to be "on" to connect one line segment to another, all the parasitic capacitances coupled to the line segment just added plus any parasitic capacitances coupled to other line segments coupled through other "on" pass transistors to the line segment just added now must be added to the parasitic capacitances coupled to the line segment already part of the interconnect being built by the software. This is a substantial complication in the software because it must do RC time constant calculations for all the new parasitics coupled to each line segment just added. This is a floating point calculation and is slower and more complicated calculation than the simple addition of a lump sum known delay of a new active repeater. Further, the new delay is unpredictable because it depends upon the previous history of the routing process where other parasitics may have been connected to the newly added line segment in previous routing operations.

In contrast, if the same length interconnect is made by coupling the same number of line segments together through the same number of active repeaters as there were pass transistors, the delay rises in only a modular, predictable fashion. The delay of the newly routed interconnect is equal to the sum of the fixed and known in advance delay of each line segment that comprises the interconnect. The delay for each segment is the total RC time constant delay for that segment alone taking into account all the parasitic capacitances affecting that line segment. The difference in routing calculation of time delays for each interconnect over the pass transistor FPGAs is that the delay calculation can be made simply by adding up the known, fixed delay of each line segment whereas in a pass transistor FPGA, the delay calculation is a floating point calculation and the outcome depends upon the previous routes which have been established. In the active repeater FPGA, the delay calculation never depends upon the previous routes that have been established. That is, the overall delay will depend upon the previous routes in the sense that the previous routes will preclude use of certain line segments that would have been used for a shorter route. Therefore, previous routing decisions may dictate how many line segments are needed to make the new route. The difference is that each new line segment's delay is known in advance, predictable and does not change with previous

routing decisions so the total delay calculation is a simple addition of the known delays for each line segment.

The reason for this behavior is that the parasitics of new line segments are being charged by the output of the active repeater coupled to that line segment using current from the Vcc supply of that active repeater and not from the source driver. In other words, with each new segment added to the interconnect, the delay rises by only the known delay of the new segment, and the load of the new segment is not added to the load of the previous segment requiring a new RC time constant calculation. Thus, higher predictability of performance can be achieved by doing interconnects through active repeaters. This not only vastly speeds up performance in terms of propagation times for signals but also substantially simplifies the calculations required of the routing software.

The same attributes apply to the biCMOS active repeaters disclosed herein. For example, with the biCMOS multiplexer of Figure 13, one line segment will be coupled to the A and A- input port which is coupled to the bases of the bipolar NPN differentially coupled devices E1 and E2 and E3 or E4. The other line segment will be coupled to either the Y1 and Y1- or Y2 and Y2- ports coupled to the collectors of the bipolar NPN differentially coupled devices E1 and E2 or E3 or E4. There is no direct conduction path for electrons to flow from the base to the collectors of the devices because any current flowing into the base is caused by recombination of carriers crossing the forward biased base-emitter junction with holes in the base region. The base current is an equivalent current to the amount of recombination occurring in the base region. Thus, current flowing into the base does not cross the reverse biased base-collector junction and reach the parasitic capacitances coupled to the line segment coupled to the collector. Instead, the parasitic capacitances coupled to the line segment coupled to the collector are charged by the main current flow supplied by the power supply which flows across the forward biased base-emitter junction, through the base region without recombination and across the reverse biased base collector junction. Thus, there is no direct current flow from the line segment coupled to the base to the line segment coupled to any of the collectors. Thus, the meaning of the phrase “no direct connection” in the claims is that there is no direct flow of electrons or holes through the active repeater from one line segment coupled to an active repeater to another line segment coupled to the active repeater. Therefore, the meaning of “active repeater” in the claims is any circuit which, in addition to being able to bring gain to bear in transferring a signal from one line segment to another in regenerating the signal by replacing losses to the parasitics with current to the power supply, but also having a structure which provides “no direct connection”

between line segments coupled to the active repeater so charge carriers cannot transition directly from one line segment to another through the active repeater.

From all of the above it is clear that pass transistor based interconnects have delays which increase at an exponential rate because of the RC networks that are added as the interconnect gets longer. Further, pass transistor based interconnects do not scale with the technology. In contrast, active repeater based interconnects have constant delay until the next repeater boundary is crossed and then the delay jumps by a known incremental amount not related to the number of parasitic capacitances coupled to the line segment added to the interconnect. Active repeater interconnects do scale with the technology.

Returning to the consideration of Figure 21, the programmable connection matrix at 370 in Figure 21 is a partially populated matrix with programmable connections, represented by bubbles or ovals, each which may be either one-way active repeater like the tri-state inverter of Figure 22A or a bidirectional active repeater like the circuit shown in Figure 23A. In some embodiments, the active repeaters within the programmable connection matrices within the core array or within the RIUs are all unidirectional active repeaters. However, in the preferred embodiment, the programmable connection matrices are primarily unidirectional active repeaters with some bidirectional active repeaters. The presence of the bidirectional active repeaters in the programmable connection matrices of the core array and the RIU's increases the number of possible connection routes that can be made by the routing software. Further, in the preferred embodiment there are different types of programmable connection matrices each with different connection possibilities. These different types of programmable connection matrices are "rolled" along the horizontal buses in the core array and "rolled" along the O-lines buses in the RIU's. "Rolled" as that term is used herein means sequentially alternated. For example, in Figure 21, if there are four different "flavors" (different connection possibilities) they will be rolled by making programmable connection matrices (PCM) 370 flavor 1, PCM 650 will be flavor 2, PCM 652 will be flavor 3, PCM 654 will be flavor 4 and PCM 656 will be flavor 1. The sequence then repeats.

This rolling is done only along the horizontal buses and not along the vertical buses. The reason for this is the rolling increases the amount of "chaos", i.e., lack of regularity, in the programmable connection possibilities available. This increased chaos eases the job of the routing software.

The different programmable connection possibilities in each PCM is not really a chaotic or random event. The particular connection possibilities for each PCM are

selected so that they are all different within the rolling group, but so as to provide "completeness" and substantially equal "weighting". "Completeness" means that the particular programmable connection possibilities selected for the individual PCM's within the rolling group are selected to be different for each PCM, but are selected such that when the rolling pattern is repeated, all logic block inputs, all logic block outputs and all I/O pins can be reached by at least one conductive path. Substantially equal "weighting" means that the programmable connection within the PCM's is selected within the rolling group of PCM's so that each PCM is different but such that when the rolling pattern is repeated in the core array and RIU, each output of a logic block or each I/O pin can be programmably connected to substantially the same number of inputs as any other logic block output or I/O pin. In other words "completeness" and substantially equal "weighting" means that in an FPGA having PCM's having these characteristics each input, output and I/O pin can be reached somehow and none are substantially choked off more than the others in terms of the routing possibilities to it.

Returning to the consideration of Figure 21, other embodiments of active repeaters disclosed herein will also work in the programmable connection matrices in the core and RIU's, but the circuits of Figures 22A and 23A are preferred. These active repeaters in matrix 370 make programmable connections between selected ones of the north-south conductors of column 350 and the conductors of east-west row 372, and, at intersections having bidirectional active repeaters, vice versa.

The logic block 356 is connected to the lines of the east-west row 372 by a bus 354 each line of which is the output of a multiplexer, the multiplexers being represented by the Xs on bus 354. The symbol 374 in this instance represents a 10 or 11 input multiplexer with its inputs coupled to each of the 10 individual conductors represented by bus 376. The multiplexer has a sufficient number of select inputs (not shown) to select one of the conductors in bus 376 which is connected to one of the inputs represented by line 354. The select signals are programmable and the logic states on these lines are defined by programming bits stored in memory cells (not shown) set by the user. The preferred circuit for the multiplexers is shown in Figure 27 which shows a 4-input buffered multiplexer which can be extended to as many inputs as needed. This same multiplexer is used for the multiplexers 456, 458 and 460 in Figure 26 and for all the multiplexers in each RIU programmable connection matrix. These buffered multiplexers are also considered active repeaters because they have gain and can reconstruct a signal and there is no direct connection between line segments coupled to the inputs and outputs.

Figure 23A is a circuit diagram of the preferred CMOS bidirectional tri-state driver, and Figure 23B is a truth table defining the operation of the circuit of Figure 23A. When the enable A (EA) and enable B (EB) programmable control inputs on lines 380 and 382 are both low, the driver is in tri-state mode and no signal on line A propagates to line B or vice versa. When EA is low and EB is high, signal line B is an input and signal line A is an output so transitions on B show up on output line A inverted and with energy lost to parasitics replaced by the active nature of the repeater. Likewise, when EA is high and EB is low, signal line A is the input and signal line B is the output so transitions on A show up inverted on B with energy lost to parasitics replaced and the signal regenerated.

The ovals within connection matrix 370 in Figure 21 represent either the one-way tri-state driver of Figure 22A or the bi-directional tri-state driver of Figure 23A. The particular programmable connection possibilities selected for connection matrix 370 and the connection matrix between the conductors of column 350 and the conductors represented by line 352 are not part of the RIU invention claimed herein and any combination of one-way and bidirectional active repeaters at any combination of junctions will suffice to practice the RIU invention.

Although the invention has been described in terms of the preferred and alternative embodiments disclosed herein, those skilled in the art will recognize various structural and functional equivalents of the circuits disclosed herein. For example, any of the differential current mode logic current switches could be replaced by single ended versions such as are shown in Figure 18. Other fast switching technology could be substituted for the bipolar devices shown herein and bipolar or other faster switching devices could be substituted for the CMOS enabling devices. All such structural and functional equivalents are intended to be included within the scope of the claims appended hereto.

What is claimed

1.. A conductor in a field programmable gate array segmented into multiple segments by active repeaters such that there is no direct conductive path between the segments.

2. A field programmable gate array having a core array of programmable logic blocks each having data inputs and data outputs which are programmably coupled to individual signal lines of at least one of a plurality of vertical and horizontal buses, said vertical and horizontal buses comprised of a plurality of individual conductive signal lines which are divided into segments by active repeaters, said vertical and horizontal buses intersecting at a plurality of intersections, each intersection being a programmable connection matrix which is sparsely populated with active repeaters which function to programmably couple preselected individual conductive signal lines of said horizontal buses to preselected individual conductive signal lines of said vertical buses for either unidirectional or bidirectional communication, sparse population being defined as the presence of an active repeater only at some but not all the intersections of individual conductive signal lines within said horizontal and vertical buses, said programmable logic blocks being coupled to each other through interconnections comprised of individual segments of said individual conductive signal lines of said vertical and horizontal buses and selected active repeaters which are programmed to couple signals between the appropriate segments to make the necessary connections to implement a desired functionality for said field programmable gate array.

3. The field programmable gate array of claim 2 further comprising a plurality of I/O cells surrounding said core array, each I/O cell being programmable as a data input or a data output, or both so as to return data emerging from said core array on a first individual conductive signal line back into the core array on a second individual conductive signal line, and a plurality of repeatable interface circuits surrounding said core array and adjacent said plurality of I/O cells and programmably and selectively coupling said individual conductive signal lines of said vertical and horizontal buses to said I/O cells.

4. The field programmable gate array of claim 3 wherein each repeatable interface circuit has a substantially identical structure and includes a routing area comprised of a plurality of programmable connection matrices which are sparsely

populated with active repeaters at selected intersections of individual signal lines, said programmable connection matrices being programmable to route signals on said individual signal conductors of said vertical and horizontal bus signal lines to specific ones of a plurality of inputs and outputs of said I/O cells.

5. The field programmable gate array of claim 2 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load.

6. The field programmable gate array of claim 3 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load.

7. The field programmable gate array of claim 4 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load.

8. A field programmable gate array having a core array of programmable logic blocks each having data inputs and data outputs which are programmably coupled to individual signal lines of at least one of a plurality of vertical and horizontal buses, said vertical and horizontal buses comprised of a plurality of individual conductive signal lines which are divided into segments by active repeaters, said active repeaters which segment the individual conductive signal lines of said vertical buses being staggered so as to increase the access of each logic block, and said active repeaters which segment the individual conductive signal lines of said horizontal buses also being staggered so as to increase the access of individual logic blocks, said vertical and horizontal buses intersecting at a plurality of programmable connection matrices, each programmable

connections matrix being sparsely populated with active repeaters which function to programmably couple preselected individual conductive signal lines of said horizontal buses to preselected individual conductive signal lines of said vertical buses, at least some of said programmable connection matrices having different connection possibilities than the others, sparse population being defined as the presence of an active repeater only at some but not all the intersections of individual conductive signal lines within said horizontal and vertical buses, different connection possibilities being defined as different patterns of sparse population from one programmable connection matrix to the next, said programmable logic blocks being coupled to each other through interconnections comprised of individual segments of said individual conductive signal lines of said vertical and horizontal buses and selected active repeaters which are programmed to couple signals between the appropriate segments to make the necessary connections to implement a desired functionality for said field programmable gate array.

9 The field programmable gate array of claim 8 further comprising a plurality of I/O cells surrounding said core array, each I/O cell being programmable as a data input or a data output or to return data emerging from said core array on a first individual conductive signal line back into the core array on a second individual conductive signal line, and a plurality of repeatable interface circuits surrounding said core array and adjacent said plurality of I/O cells and programmably and selectively coupling said individual conductive signal lines of said vertical and horizontal buses to said I/O cells, and wherein the different programmable connection matrices having a rolling pattern along said horizontal buses only.

10. The field programmable gate array of claim 9 wherein each repeatable interface circuit has an substantially identical structure and includes a routing area comprised of a plurality of programmable connection matrices which are sparsely populated with active repeaters at selected intersections of individual signal lines, and wherein at least some of said programmable connection matrices within each repeatable interface circuit have different connection possibilities through their sparse populations of active repeaters, said programmable connection matrices being programmable to route signals on said individual signal conductors of said vertical and horizontal bus signal lines to specific ones of said I/O cells.

11. The field programmable gate array of claim 8 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load.

12. The field programmable gate array of claim 9 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load, and wherein each line segment bounded by active repeaters has approximately the same parasitic capacitance as other line segments.

13. The field programmable gate array of claim 10 wherein each programmable logic block has data outputs that are programmably coupled to selected individual signal conductors of at least one vertical or horizontal bus by drivers, said drivers being optimally structured in integrated circuit form to drive a known capacitive load, and wherein each active repeater is optimally structured in integrated circuit form to drive a known capacitive load.

14. A process for making interconnections in a field programmable gate array having an array of programmable logic blocks programmably coupled through drivers and multiplexers to a plurality of individual signal conductors in buses which intersect each other, each individual signal conductor being divided into segments by active repeaters which can be programmed to either connect or not connect the two line segments coupled to each active repeater, and said intersections of said buses being partially populated with unidirectional and at least some bidirectional active repeaters in the form of tri-state drivers such that selected lines in each bus can be programmable connected to selected lines in the intersecting bus, comprising forming an interconnection by programming the appropriate bidirectional and unidirectional active repeaters to connect together the appropriate line segments to form the interconnection, different line segments having different, known-in-advance, parasitic capacitance loads which do not change when different interconnecting pathways are formed.

15. A field programmable gate array apparatus having an array of programmable logic blocks and a plurality of I/O cells, and further comprising:

a plurality of horizontal and vertical buses each comprised of a plurality of data input and data output signal lines which are each segmented into a plurality of segments by active repeaters which can be programmed to be "on" or "off";

a plurality of active repeaters structured to selectively programmably couple predetermined individual conductors of said horizontal buses to predetermined individual conductors of said vertical buses at a plurality of locations; and

a plurality of repeatable interface units coupling said I/O cells to said data input and data output signal lines through routing areas comprised of a plurality of O_lines and a plurality of I_lines and a plurality of active repeaters.

16. A conductive line in a field programmable gate array divided into segments by a plurality of active repeater means, each active repeater means for selectively, programmably coupling one line segment to another and for applying gain to a signal propagating in one line segment so as to reconstruct said signal and replace energy lost to parasitics before outputting the reconstructed signal on another line segment.

17. A field programmable gate array apparatus having an array of programmable logic blocks and a plurality of I/O cells, and further comprising:

a plurality of horizontal and vertical buses each comprised of a plurality of data input and data output signal lines which are each segmented into a plurality of segments by bidirectional active repeater means, each said bidirectional active repeater means for selectively, programmably coupling one line segment to another and for applying gain to a signal propagating in one line segment so as to reconstruct said signal and replace energy lost to parasitics before outputting the reconstructed signal on another line segment and perform these functions for signals propagating in either direction;

a plurality of active repeater means for selectively programmably coupling predetermined individual conductors of said horizontal buses to predetermined individual conductors of said vertical buses at a plurality of locations and for applying gain to a signal propagating in one line segment so as to reconstruct said signal and replace energy lost to parasitics before outputting the reconstructed signal on another line segment; and

a plurality of repeatable interface units coupling said I/O cells to said data input and data output signal lines through routing areas comprised of a plurality of O_line segments and a plurality of I_line segments and a plurality of programmable connection matrices sparsely populated with active repeater means to a plurality of selected line segments of either said vertical or horizontal buses and a plurality of line segments going into and coming out of said I/O cells, each said active repeater means for selectively, programmably coupling one line segment to another and for applying gain to a signal propagating in one line segment so as to reconstruct said signal and replace energy lost to parasitics before outputting the reconstructed signal on another line segment.

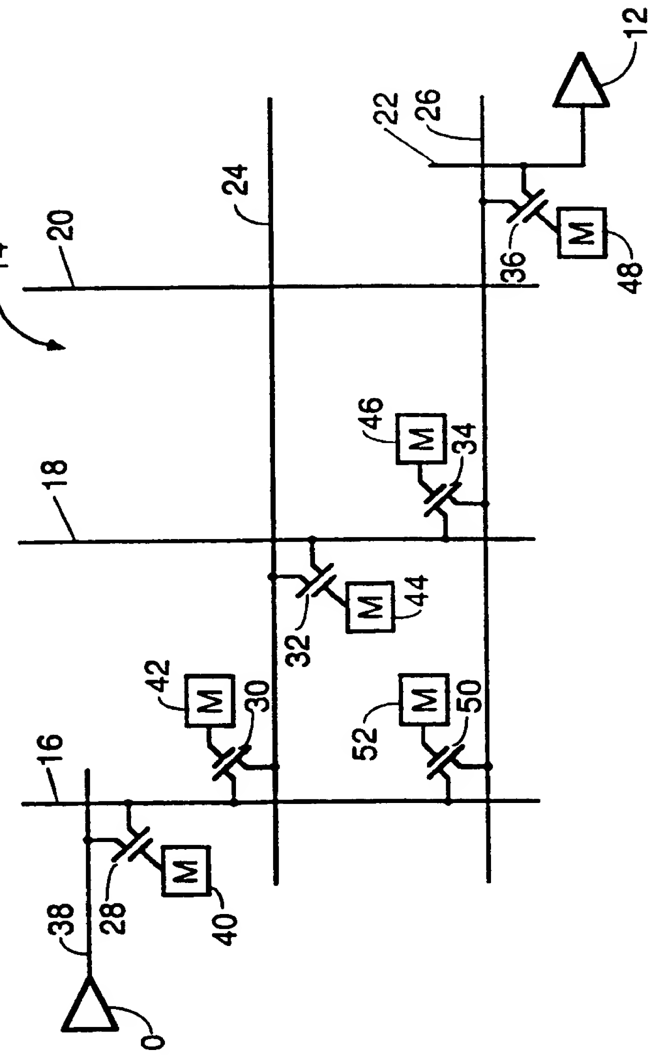
18. The apparatus of claim 17 further comprising bidirectional active repeater means coupling both ends of all or a preselected number of said O_lines and said I_lines of each repeatable interface unit to the corresponding O_lines and I_lines of neighboring repeatable interface units.

19. A field programmable gate array having a plurality of programmable I/O cells and a plurality of programmable logic blocks and a plurality of data input and data output signal lines forming signal buses and which are programmably coupled to said logic blocks, characterized by a plurality of repeatable interface units coupling said I/O cells to said data input and data output signal lines through routing areas comprised of a plurality of O_line segments and a plurality of I_line segments and a plurality of line segments coupled to said data input and data output signal lines of said signal buses and a plurality of line segments going into and coming out of said I/O cells and a plurality of active repeater means, each said active repeater means for selectively, programmably coupling one line segment to another and for applying gain to a signal propagating in one line segment so as to reconstruct said signal and replace energy lost to parasitics before outputting the reconstructed signal on another line segment.

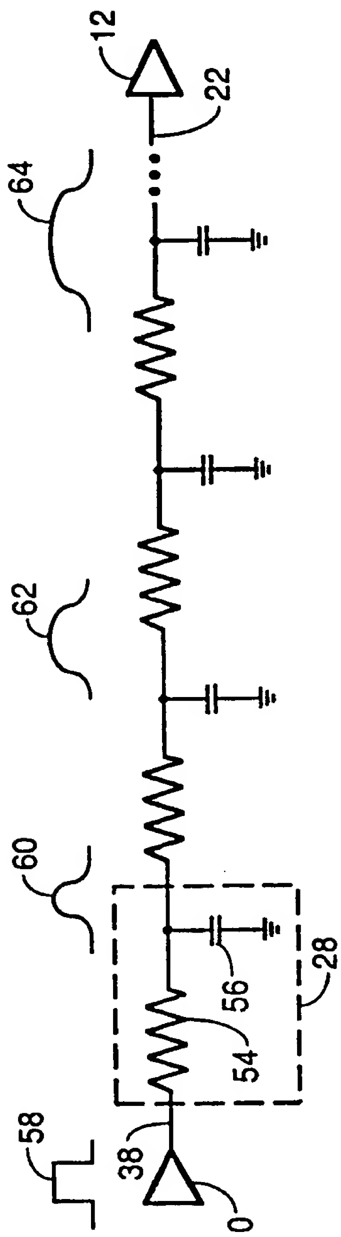
20. The apparatus of claim 19 further comprising a plurality of bidirectional active repeaters selectively, programmably coupling at least some of said O_line segments of at least some of said repeatable interface units to at least some of the O_line segments of one or more neighboring repeatable interface units, and further comprising a plurality of bidirectional active repeaters selectively, programmably coupling at least some of said I_line segments of at least some of said repeatable interface units to at least some of the I_line segments of one or more neighboring repeatable interface units.

21. A field programmable gate array having a plurality of programmable I/O cells and a plurality of programmable logic blocks and a plurality of data input and data output signal lines forming signal buses and which are programmably coupled to said logic blocks and said buses intersecting at partially populated programmable connection matrices, characterized by a plurality of unidirectional and bidirectional active repeater means in said programmable connection matrices, each said active repeater means for selectively, programmably coupling one signal line to another and for limiting the amount of parasitic capacitive load which loads a driver coupled to each line segment between active repeater boundaries to only the total parasitic capacitances actually directly connected to said line segment.

(PRIOR ART)
FIG. 1



(PRIOR ART)
FIG. 2



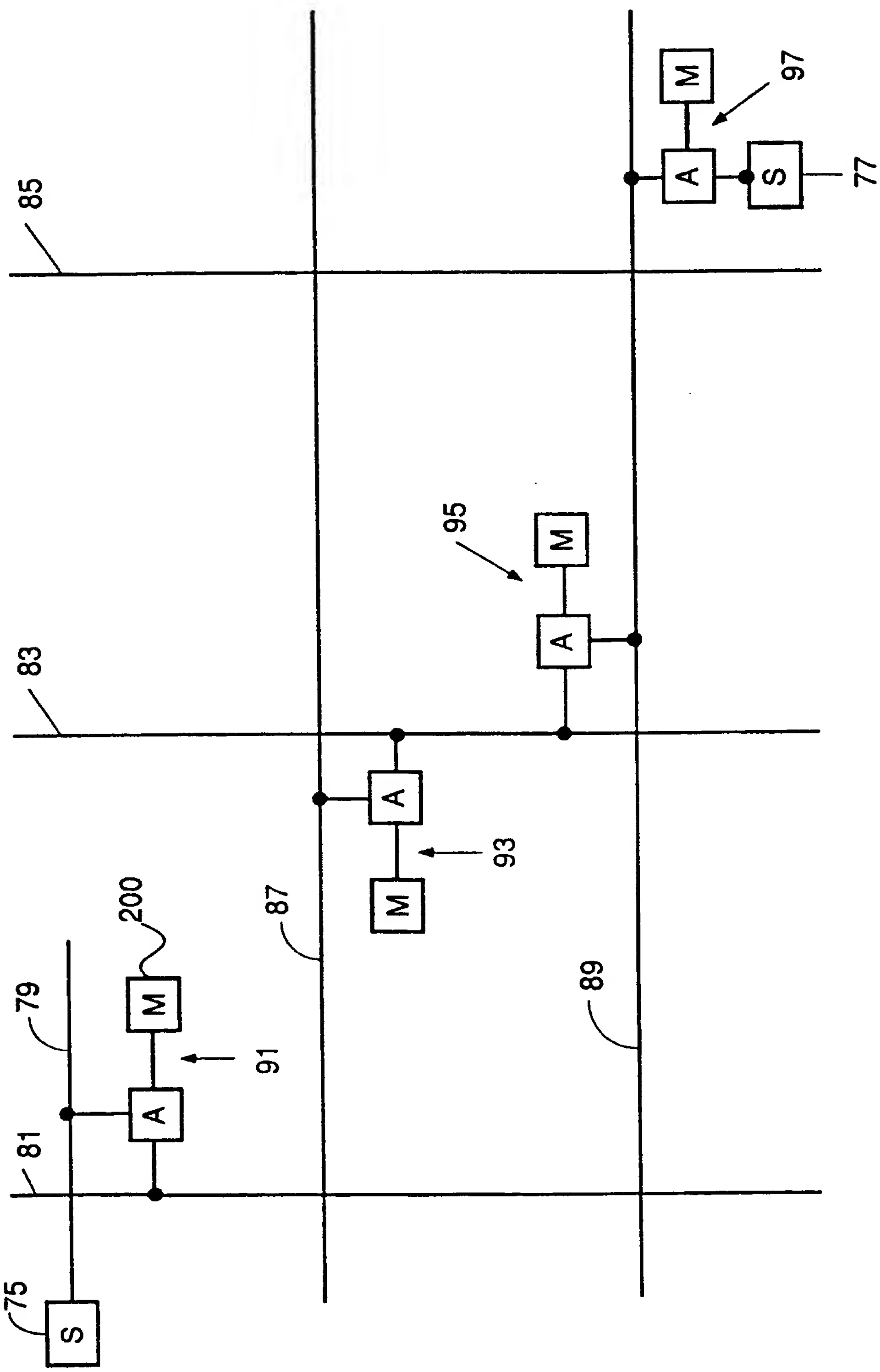


FIG. 5

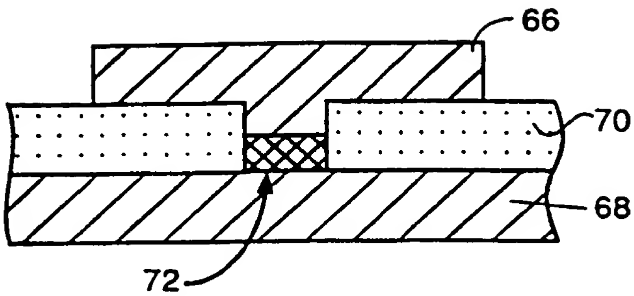


FIG. 3
(PRIOR ART)

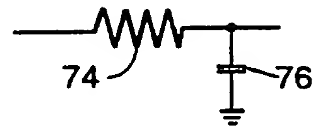


FIG. 4
(PRIOR ART)

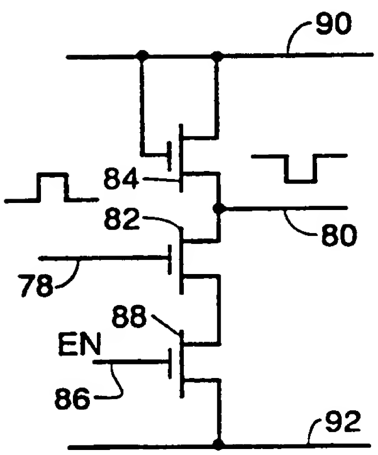


FIG. 6A

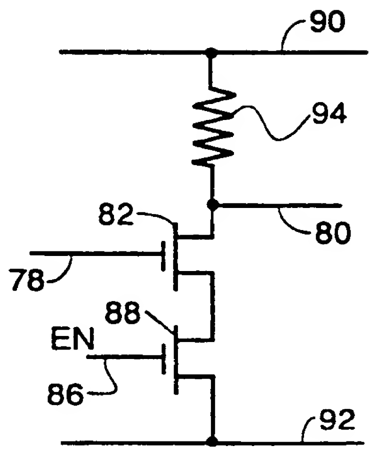


FIG. 6B

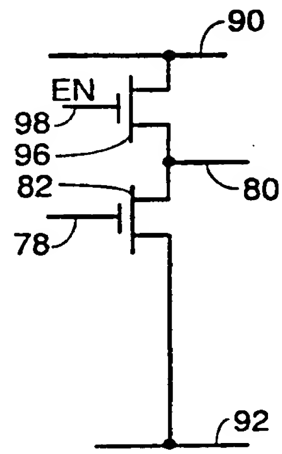


FIG. 6C

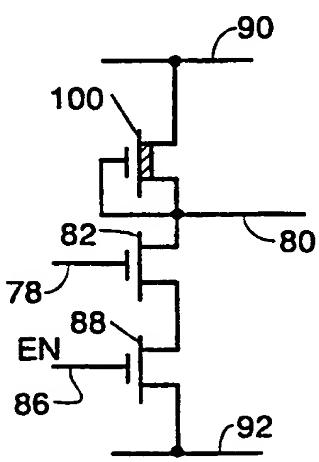


FIG. 6D

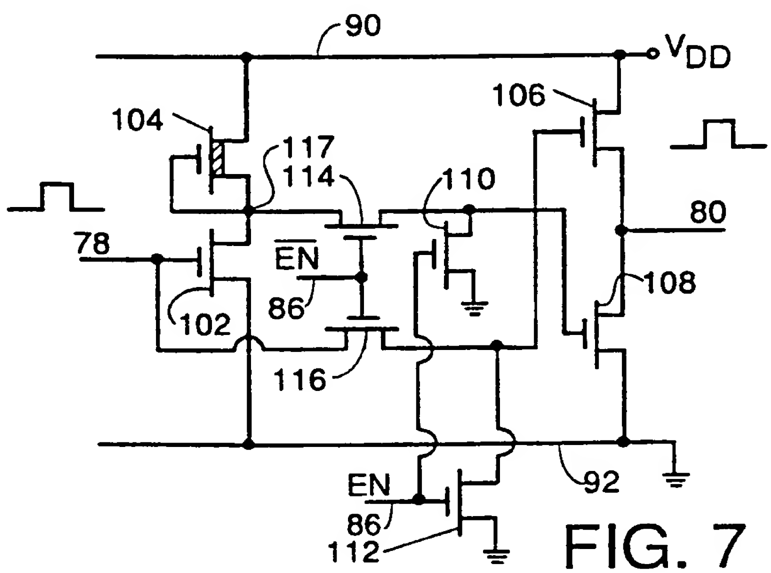


FIG. 7

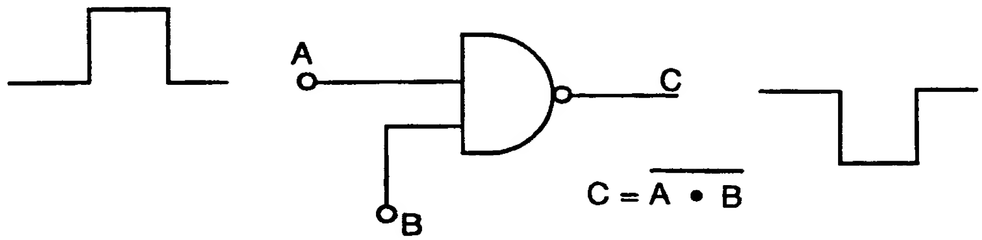


FIG. 8

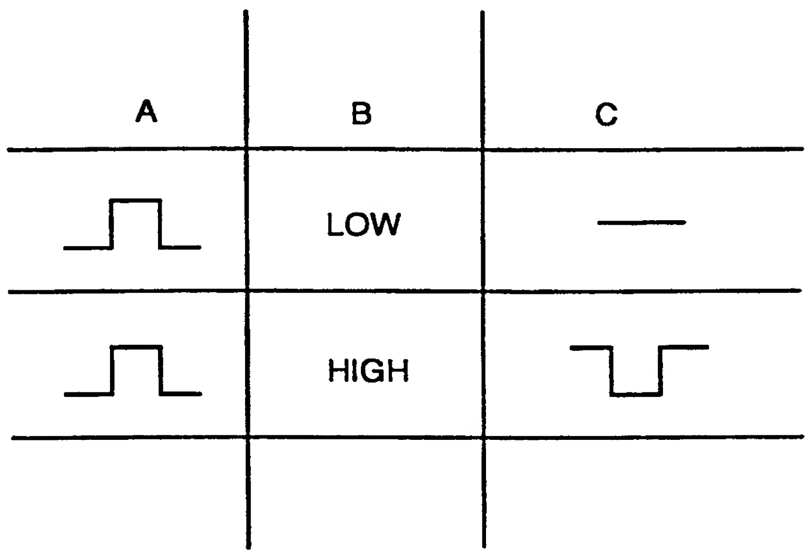


FIG.9

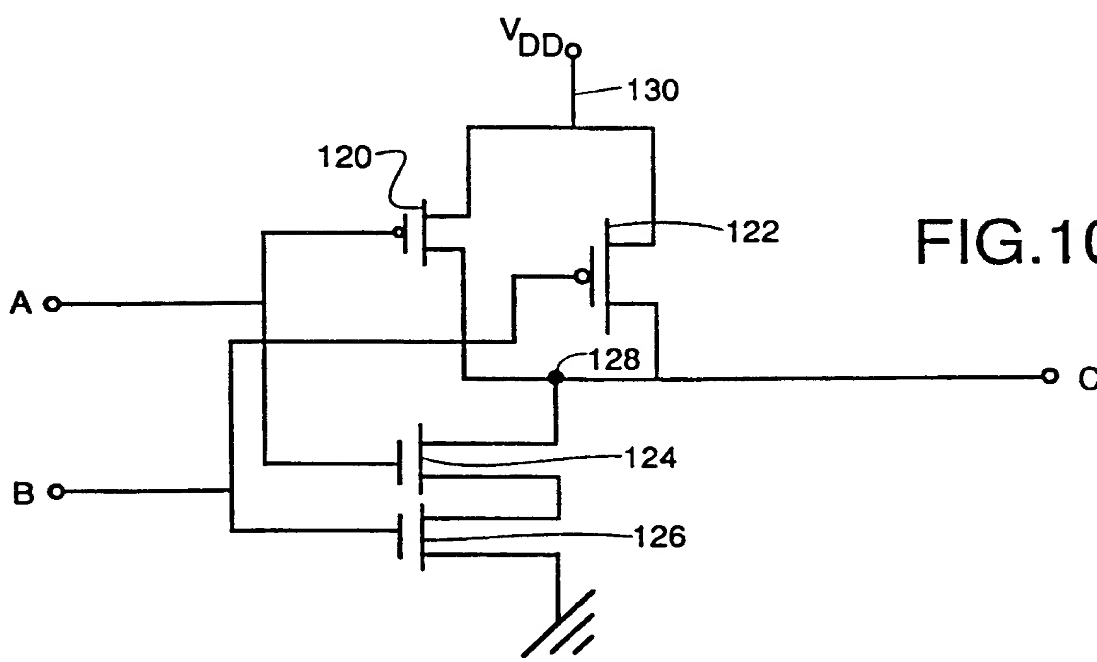
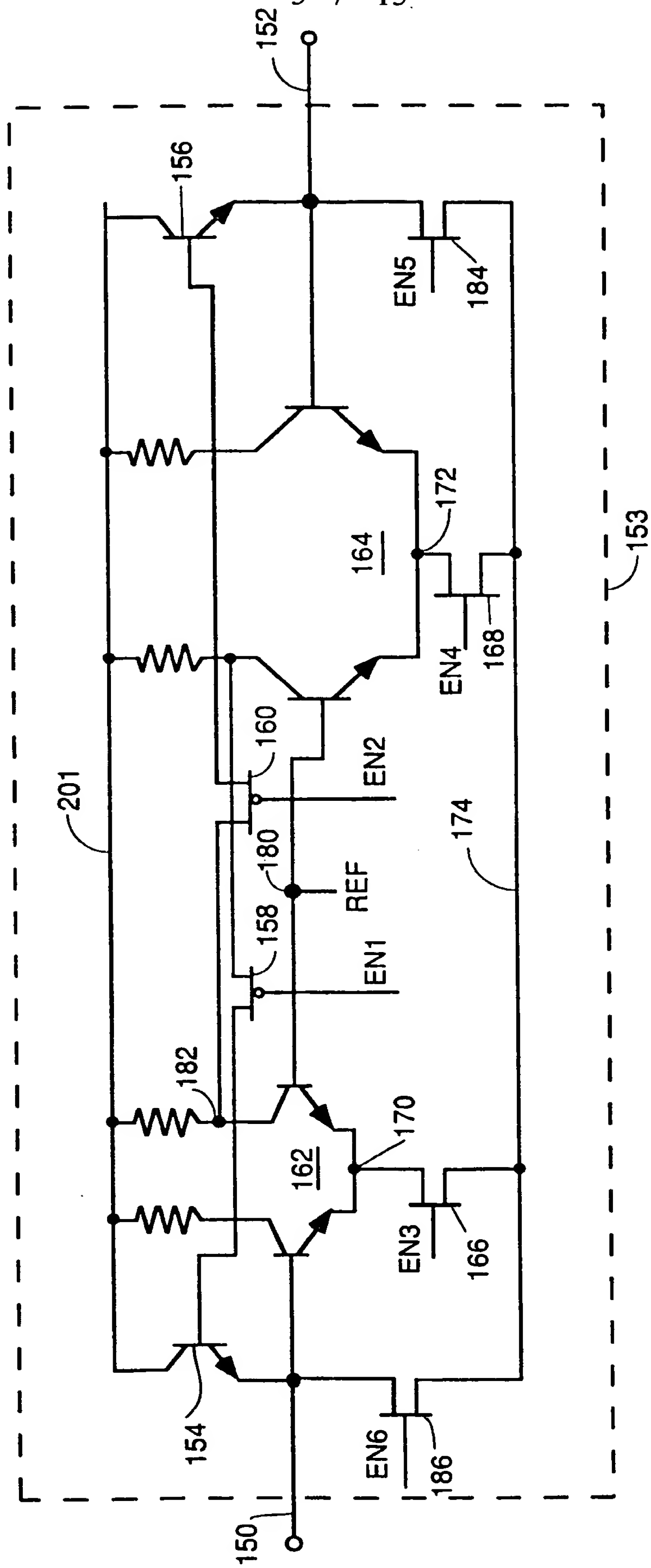


FIG.10



152
153
156
184
EN5
164
172
168
EN4
160
EN2
174
180
REF
158
EN1
201
182
170
162
166
EN3
154
186
EN6
150

FIG. 11

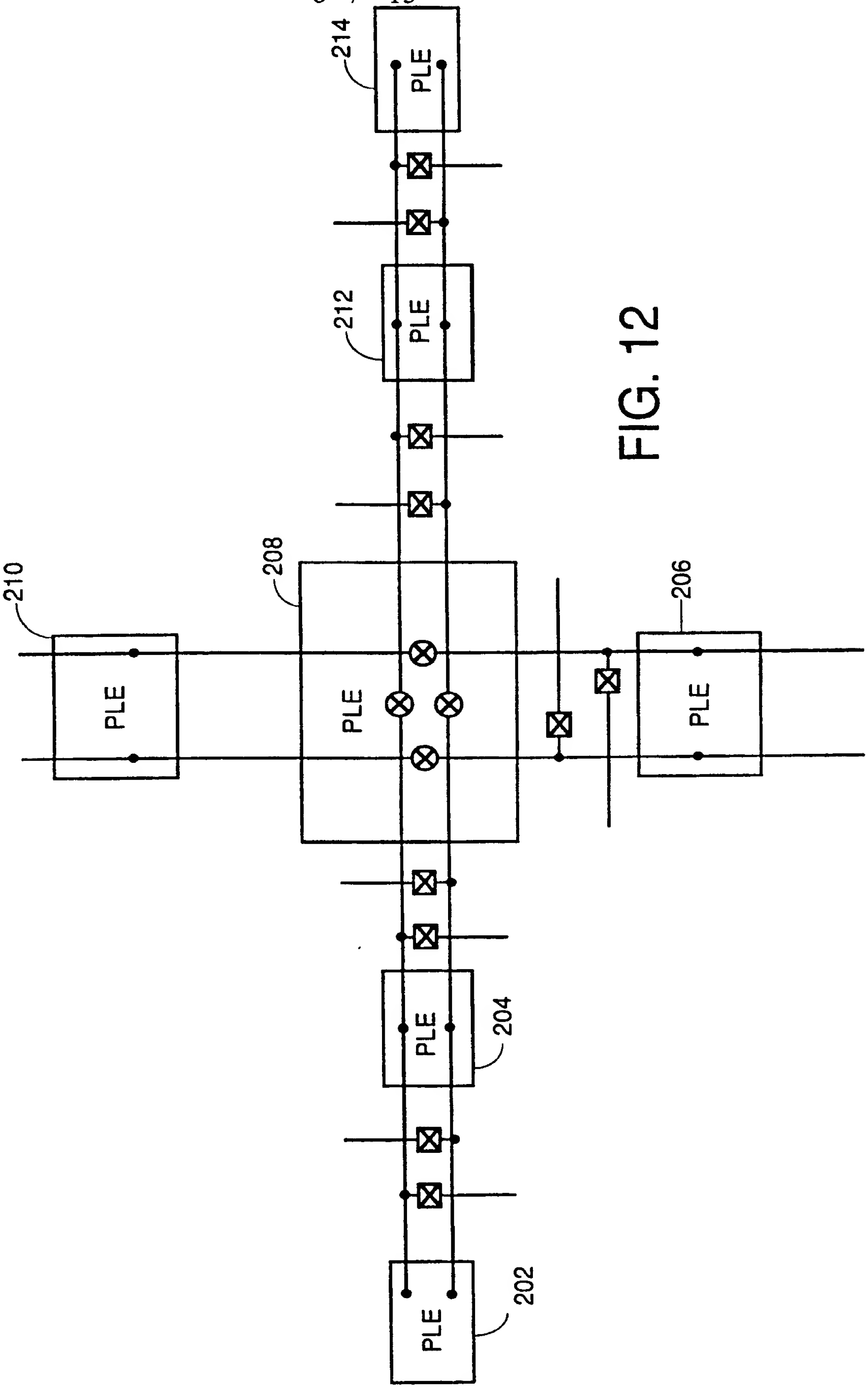


FIG. 12

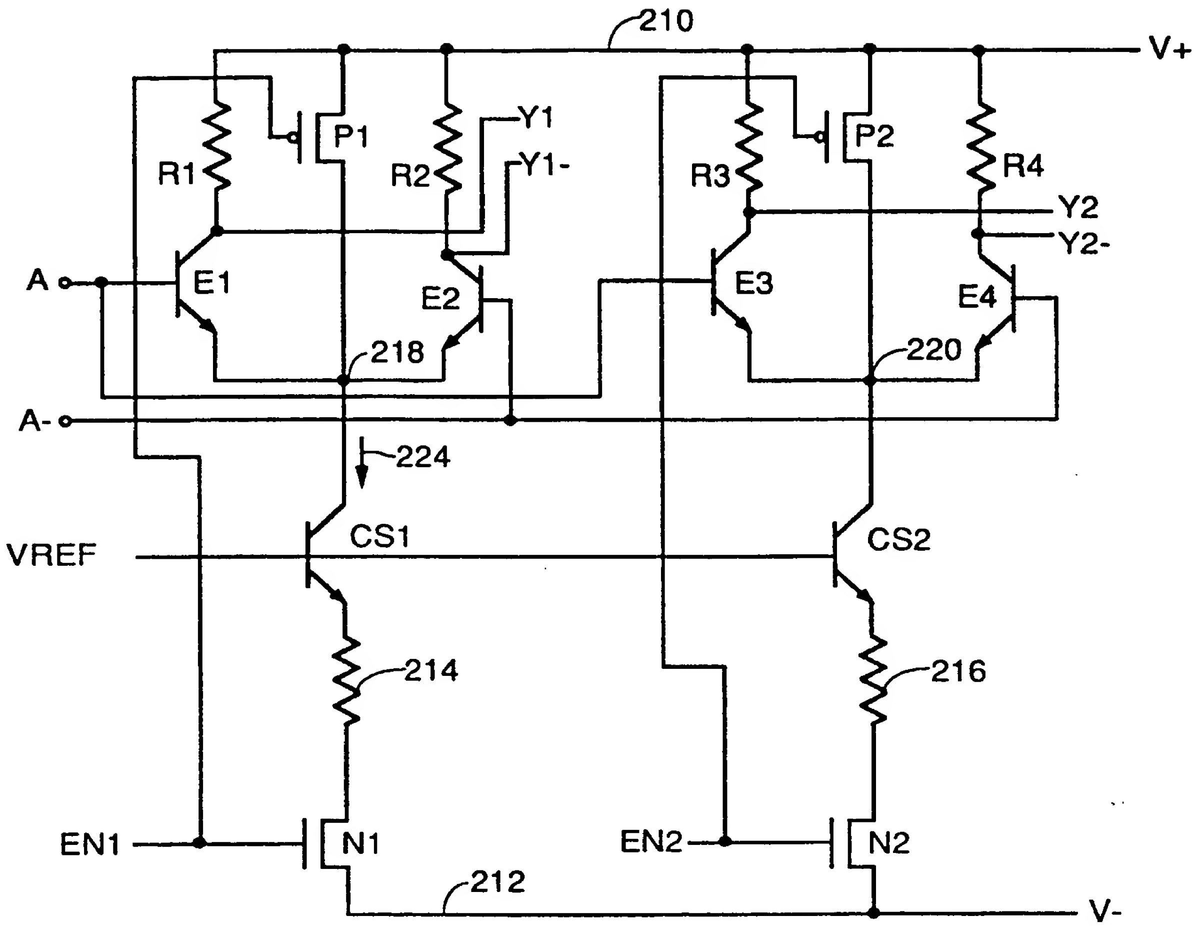

210
V+
R1
P1
R2
Y1
Y1-
R3
P2
R4
Y2
Y2-
A
E1
E2
E3
E4
218
220
A-
224
VREF
CS1
CS2
214
216
EN1
N1
EN2
N2
212
V-

FIG. 13

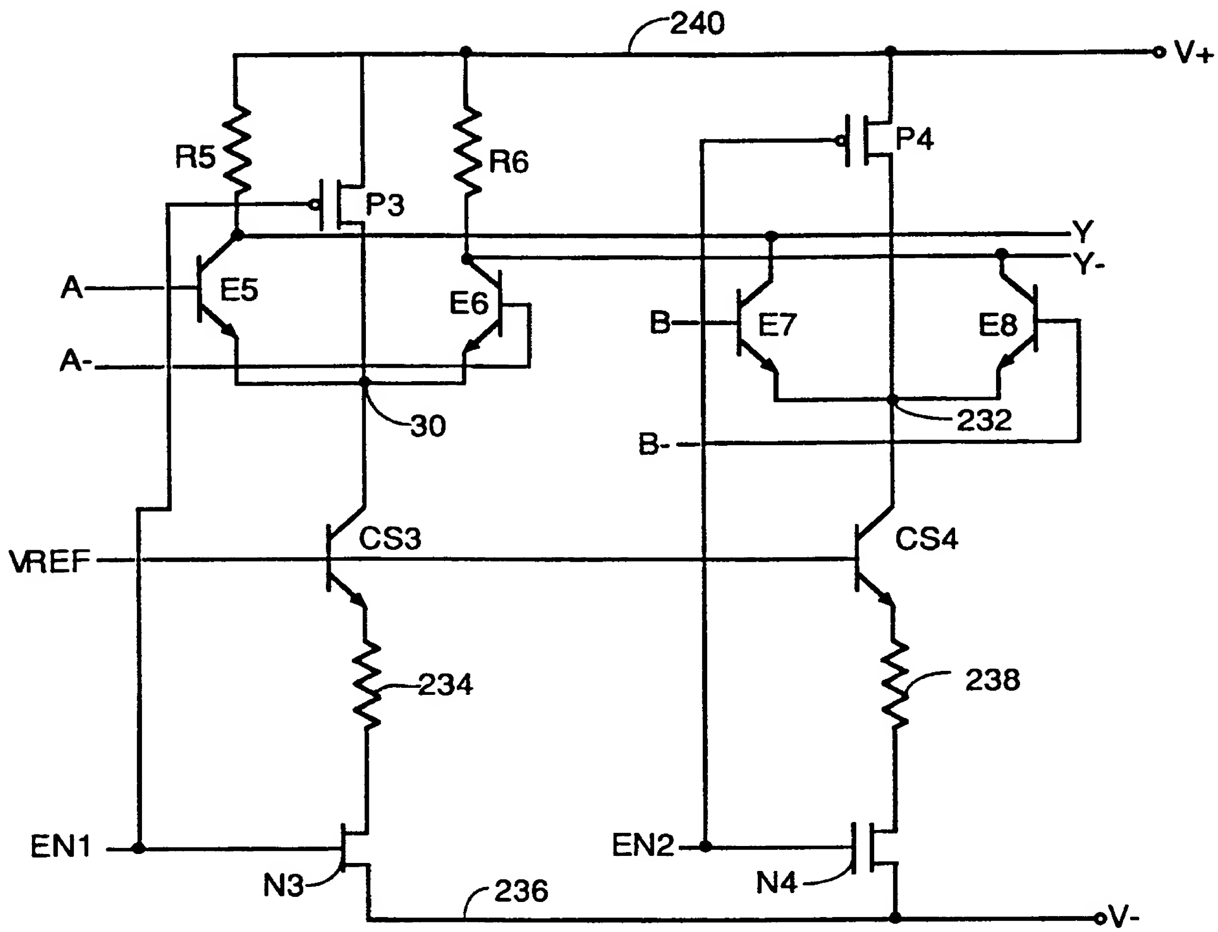

240
V+
R5
P3
R6
P4
Y
Y-
A
E5
E6
B
E7
E8
A-
30
232
B-
CS3
CS4
VREF
234
238
EN1
EN2
N3
236
N4
V-

FIG. 14

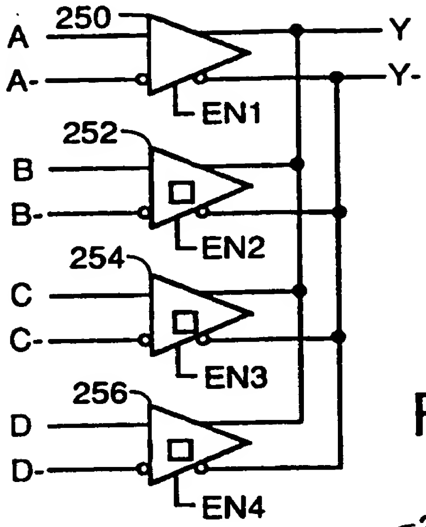


FIG. 15

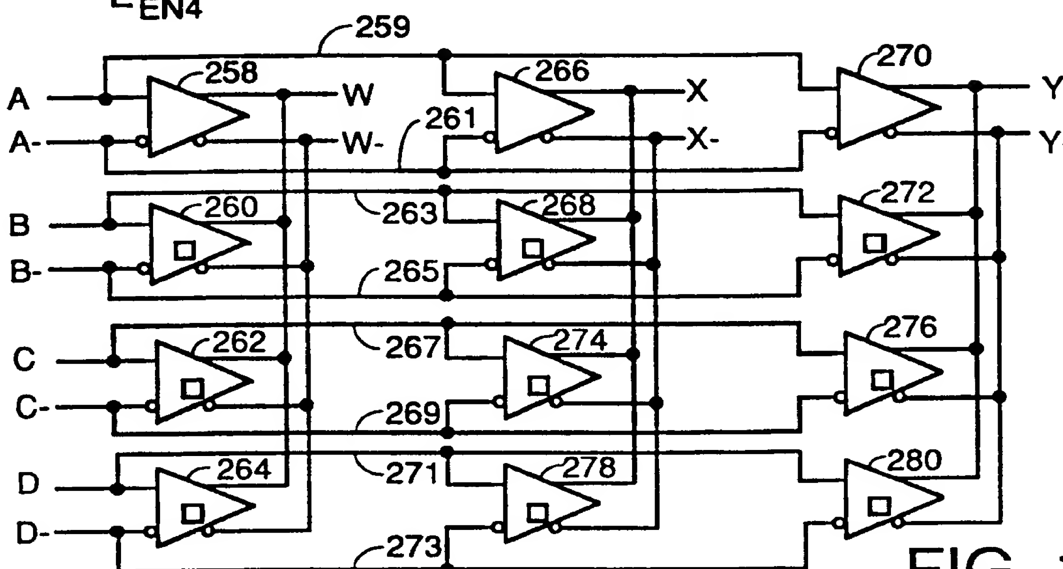


FIG. 16

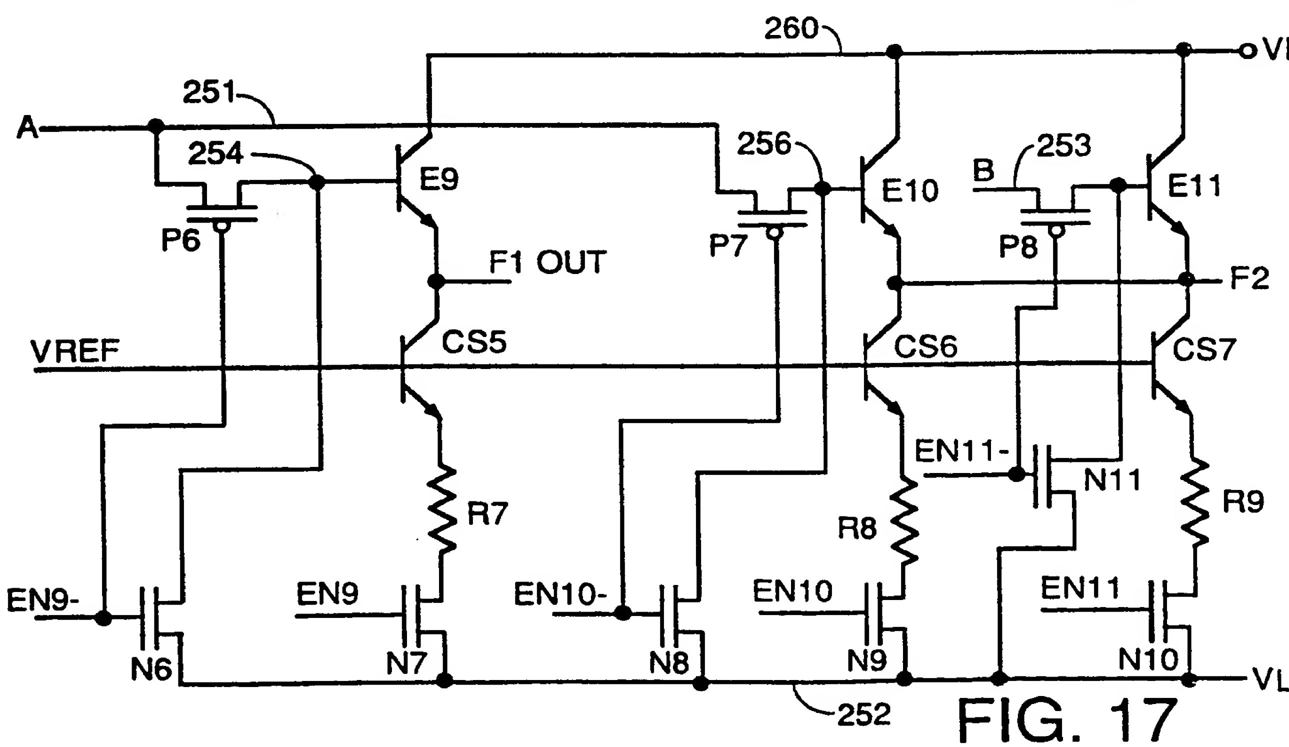


FIG. 17

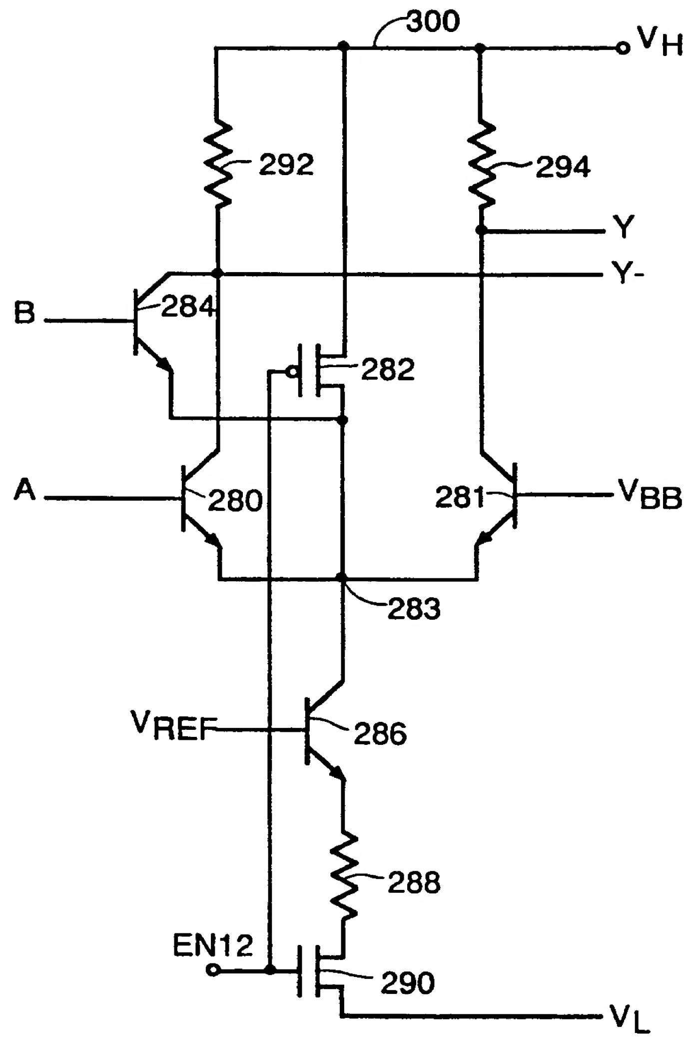

300
VH
292
294
Y
Y-
B
284
282
A
280
281
VBB
283
VREF
286
288
EN12
290
VL

FIG. 18

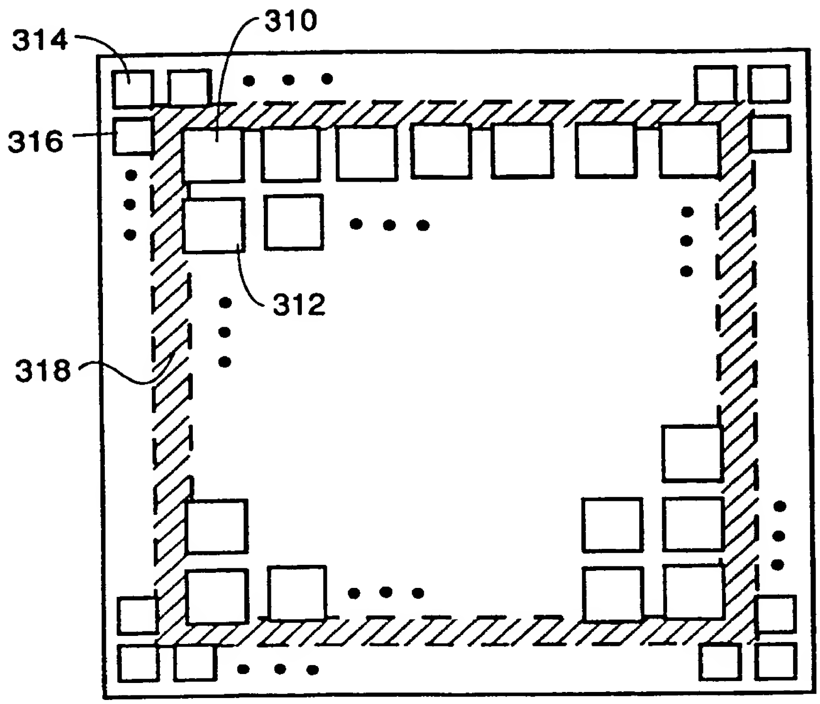

314
310
316
312
318

FIG. 19

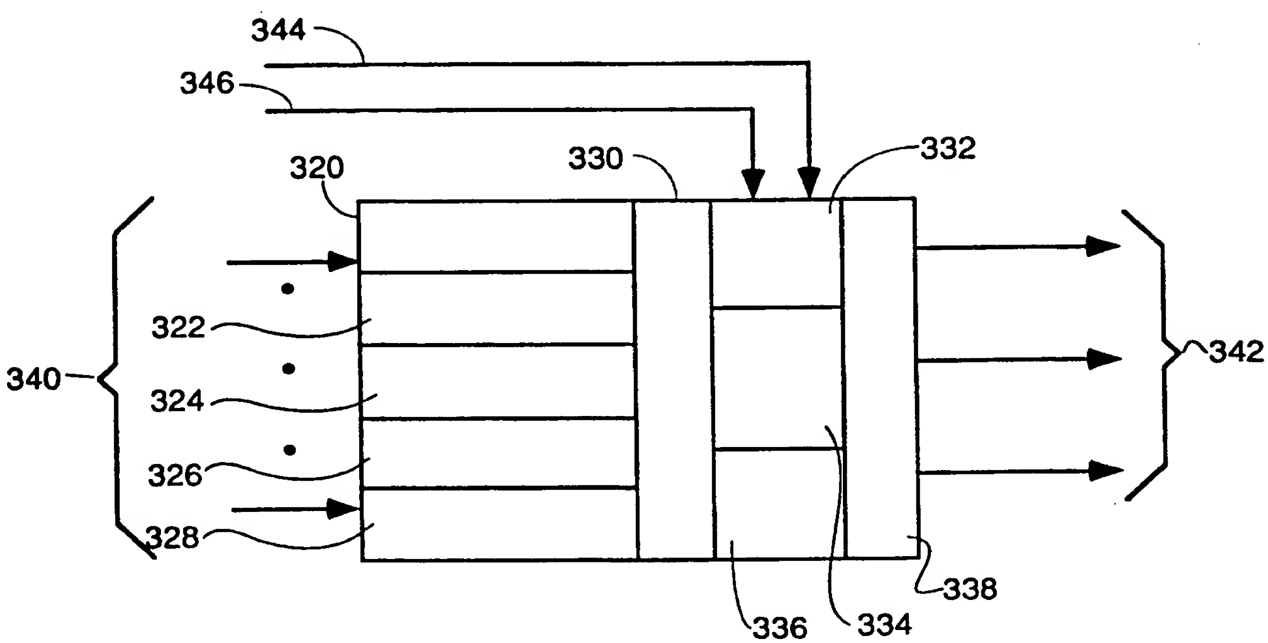

344
346
320
330
332
322
340
324
326
328
342
336
334
338

FIG. 20

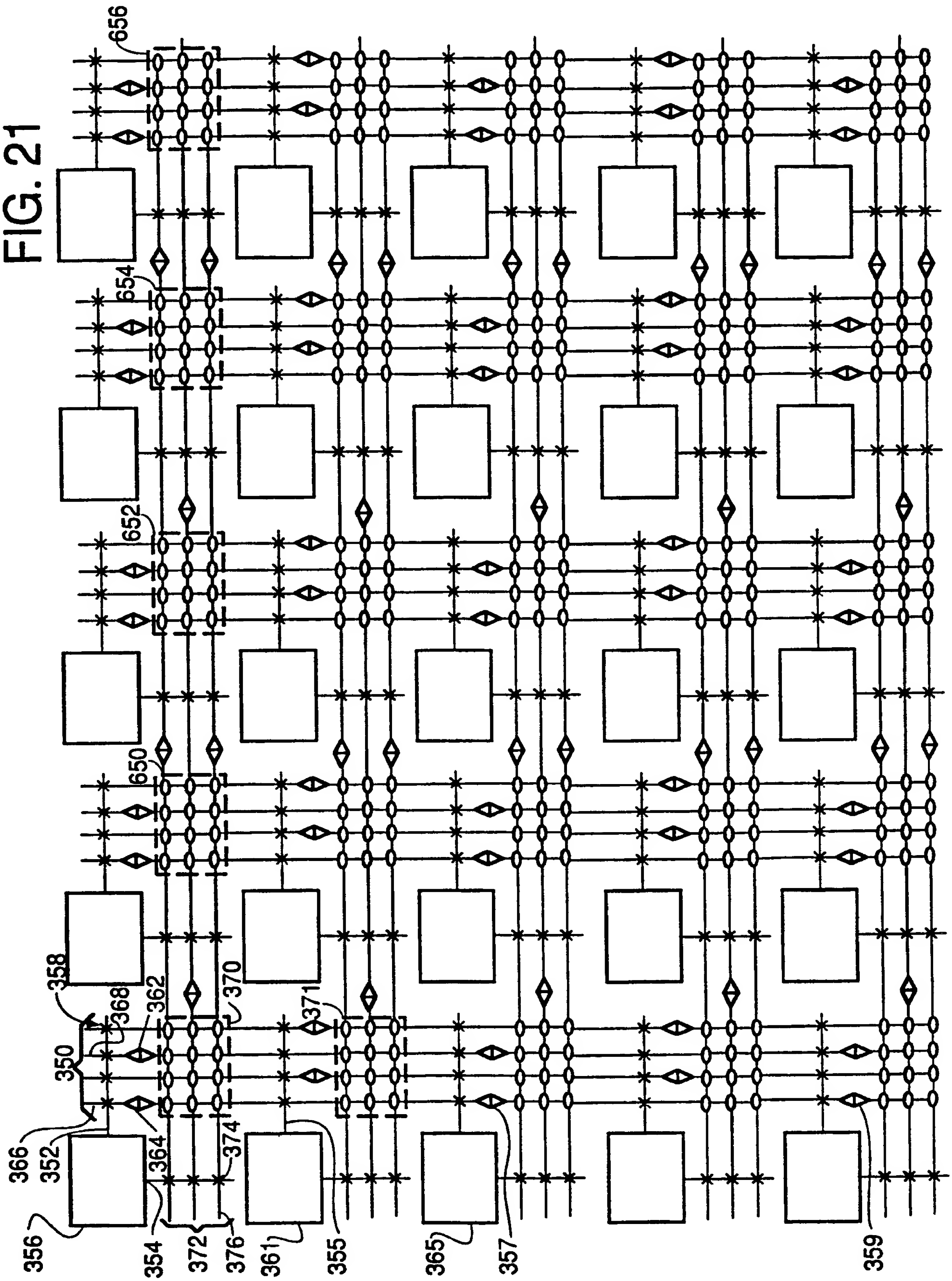

656
654
652
650
358
368
362
370
371
350
366
352
364
374
356
354
372
376
361
355
365
357
359

FIG. 21

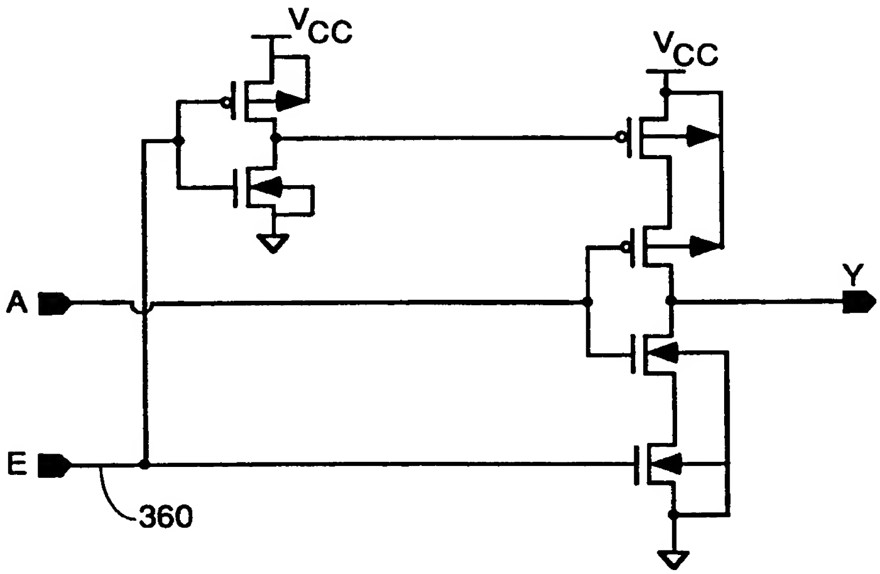


FIG. 22A

E	A	Y
0	0	Z
0	1	Z
1	0	1
1	1	0

FIG. 22B

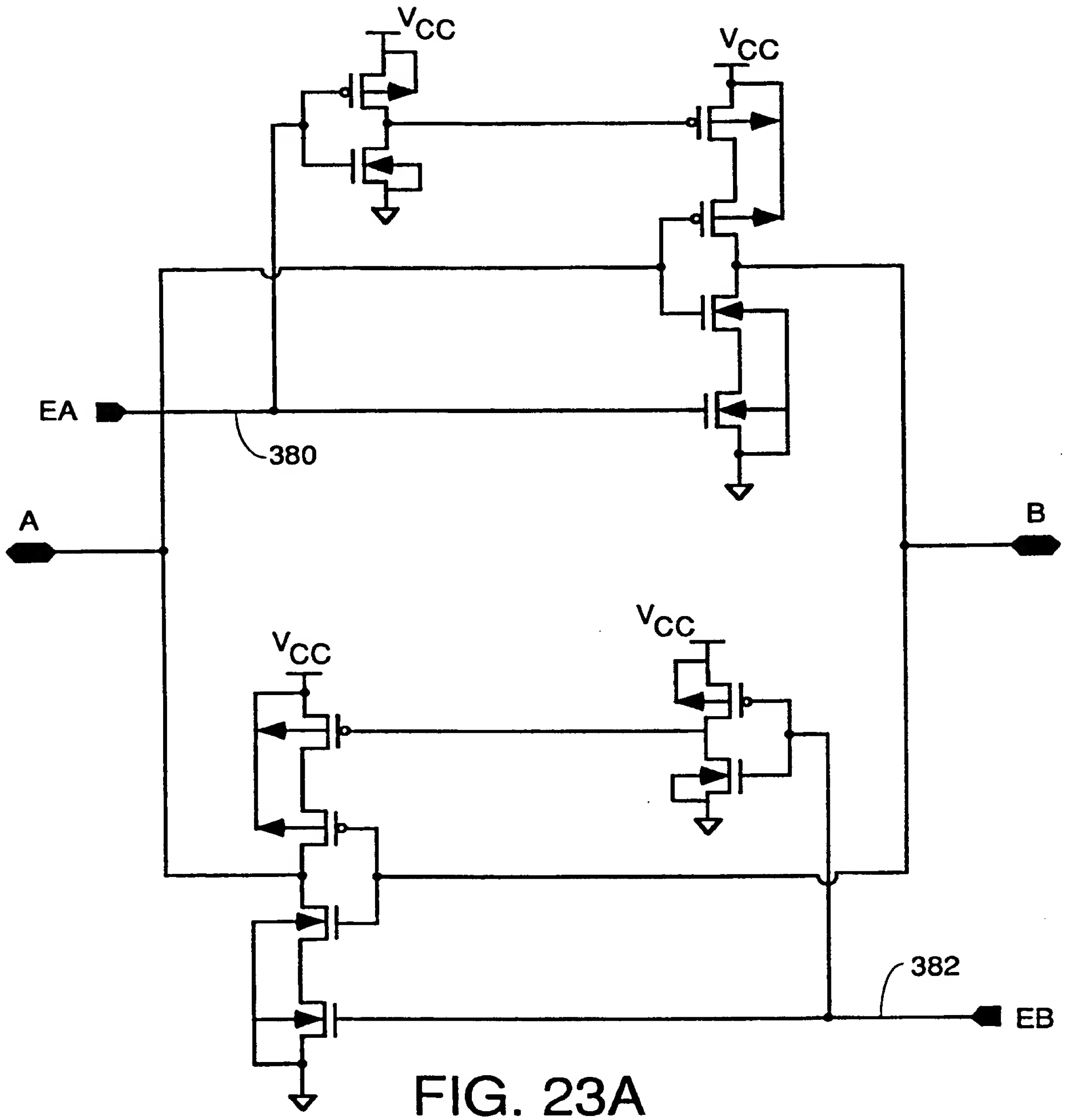


FIG. 23A

EA	EB	A	B
0	0	Z	Z
0	1	$\overline{B}$	B
1	0	A	$\overline{A}$
1	1	NOT ALLOWED	

FIG. 23B

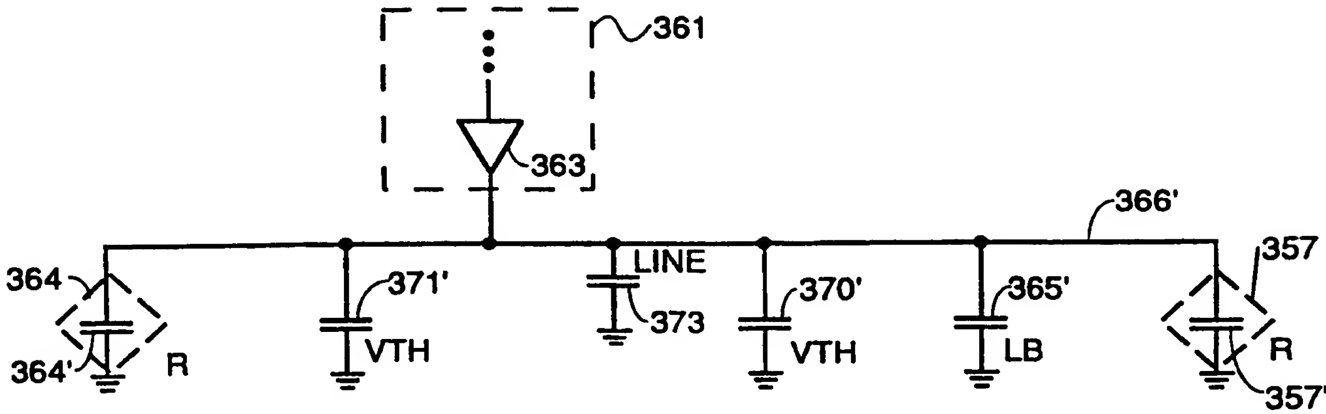

361
363
366'
364
364'
R
371'
VTH
LINE
373
370'
VTH
365'
LB
357
R
357'

FIG. 24

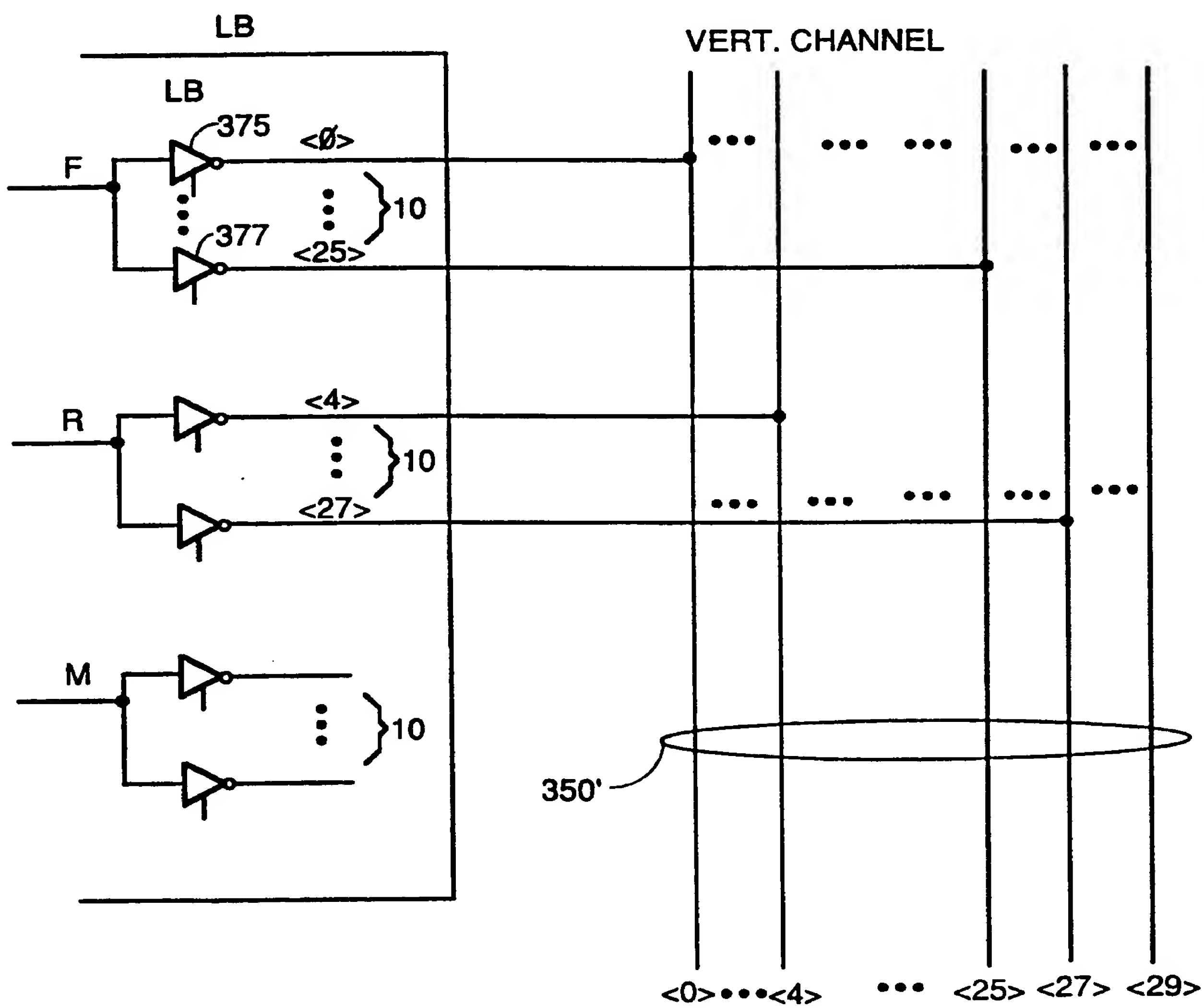

LB
LB
375
F
<Ø>
10
377
<25>
R
<4>
10
<27>
M
10
VERT. CHANNEL
350'
<0>•••<4>
•••
<25>
<27>
<29>

FIG. 25

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/11440

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : G06F 7/38, H01L 25/00, H03K 19/177
US CL : 326/38, 39, 40, 41
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 326/38, 39, 40, 41

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS (US PTO)
search terms: FPGA, link, repeater

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 5,543,640 A (SUTHERLAND ET AL) 06 August 1996 (06.08.96), see entire document.	1-4, 8, 15-20 ---------- 5-7, 9-14, 21
Y	US 5,512,765 A (GAVERICK) 30 April 1996 (30.04.96), see entire document.	1-21
Y	US 5,692,147 A (LARSEN ET AL) 25 November 1997 (25.11.97), see entire document.	1-21
Y	US 5,671,432 A (BERTOLET ET AL) 23 September 1997 (23.09.97), see entire document.	1-21
Y	US 5,631,578 A (CLINTON ET AL) 20 May 1997 (20.05.97), see entire document.	1-21

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
17 JULY 1998	19 AUG 1998

Name and mailing address of the ISA/US	Authorized officer
Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	for DANIEL CHANG
Facsimile No. (703) 305-7724	Telephone No. (703) 306-4549

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US98/11440

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,552,721 A (COULD) 03 September 1996 (3.09.96), see entire document.	1-21
Y	US 5,497,108 A (MENON ET AL) 05 March 1996 (05.03.96), see entire document.	1-21
Y	US 5,504,440 A (SASAKI) 02 April 1996 (02.04.96), see entire document.	1-21